

UNIVERSIDADE DE SÃO PAULO
ESCOLA DE ENGENHARIA DE SÃO CARLOS

Tiago Xavier Silva

**Robustez de arquitetura fotônica baseada no paradigma
Reservoir Computing para inteligência artificial**

São Carlos

2024

Tiago Xavier Silva

**Robustez de arquitetura fotônica baseada no paradigma
Reservoir Computing para inteligência artificial**

Monografia apresentada ao Curso de Engenharia Elétrica com Ênfase em Eletrônica, da Escola de Engenharia de São Carlos da Universidade de São Paulo, como parte dos requisitos para obtenção do título de Engenheiro Eletricista.

Orientador: Prof. Dr. Ben Hur Viana Borges

São Carlos

2024

AUTORIZO A REPRODUÇÃO TOTAL OU PARCIAL DESTE TRABALHO, POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

Ficha catalográfica elaborada pela Biblioteca Prof. Dr. Sérgio Rodrigues Fontes da EESC/USP com os dados inseridos pelo(a) autor(a).

S586r Silva, Tiago Xavier
Robustez de arquitetura fotônica baseada no paradigma Reservoir Computing para inteligência artificial / Tiago Xavier Silva; orientador Ben Hur Viana Borges . São Carlos, 2024.

Monografia (Graduação em Engenharia Elétrica com ênfase em Eletrônica) -- Escola de Engenharia de São Carlos da Universidade de São Paulo, 2024.

1. reservoir computing. 2. microrressonadores em anel. 3. robustez. 4. variação de temperatura. 5. computação neuromórfica fotônica. 6. machine learning.
I. Título.

FOLHA DE APROVAÇÃO

Nome: Tiago Xavier Silva

Título: “Robustez de arquitetura fotônica baseada no paradigma Reservoir Computing para inteligência artificial”

**Trabalho de Conclusão de Curso defendido e aprovado em
06/12/2024,**

com NOTA_10,0_(DEZ, ZERO), pela Comissão Julgadora:

Prof. Titular Ben-Hur Viana Borges - Orientador SEL/EESC/USP

Prof. Dr. Vinicius Marrara Pepino - SEL/EESC/USP

Prof. Dr. Gregório Couto Faria - FCM/IFSC/USP

**Coordenador da CoC-Engenharia Elétrica - EESC/USP:
Professor Associado José Carlos de Melo Vieira Júnior**

RESUMO

SILVA, T. X. **Robustez de arquitetura fotônica baseada no paradigma *Reservoir Computing* para inteligência artificial.** 2024. 73 p. Monografia (Trabalho de Conclusão de Curso) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2024.

Atualmente, técnicas de *machine learning* têm sido amplamente utilizadas para o processamento de grandes volumes de dados. Este trabalho investiga a interface entre *machine learning*, computação neuromórfica fotônica e o paradigma de *reservoir computing*, propondo a análise de microrressonadores em anel na configuração *all-pass* como reservatórios fotônicos. Por meio de uma prova de conceito, demonstra-se como essas estruturas podem realizar tarefas computacionais inspiradas no funcionamento de redes neurais, destacando sua viabilidade no processamento eficiente de dados em ambientes fotônicos. Adicionalmente, o comportamento do sistema foi analisado frente a variações térmicas, com ênfase na robustez e no potencial para aplicações práticas. Os resultados de simulação evidenciam igualmente o impacto de parâmetros de projeto na performance do reservatório, buscando contribuir com insights para o avanço da computação neuromórfica fotônica.

Palavras-chave: machine learning, computação neuromórfica fotônica, reservoir computing, microrressonadores em anel, robustez, variação de temperatura

ABSTRACT

SILVA, T. X. **Robustness of photonic systems based on Reservoir Computing approach for artificial intelligence**. 2024. 73 p. Monograph (Conclusion Course Paper) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2024.

Currently, machine learning techniques have been widely employed for processing large volumes of data. This work investigates the interface between machine learning, photonic neuromorphic computing, and the reservoir computing paradigm, proposing the analysis of ring microrresonators in the all-pass configuration as photonic reservoirs. Through a proof of concept, it is demonstrated how these structures can perform computational tasks inspired by the functioning of neural networks, highlighting their feasibility for efficient data processing in photonic environments. Additionally, the system's behavior was analyzed under thermal variations, with an emphasis on robustness and potential for practical applications. Simulation results also highlight the impact of design parameters on reservoir performance, aiming to provide insights for advancing photonic neuromorphic computing.

Keywords: machine learning, photonic neuromorphic computing, reservoir computing, ring microrresonators, robustness, temperature variation

SUMÁRIO

1	INTRODUÇÃO	11
1.1	Fotônica em silício	13
1.2	Computação neuromórfica e <i>reservoir computing</i> fotônicos	14
1.3	Objetivos	15
2	FUNDAMENTOS DE MACHINE LEARNING	17
2.1	Tipos de problemas em <i>machine learning</i>	17
2.1.1	Aprendizado supervisionado	17
2.1.1.1	Regressão	18
2.1.1.2	Classificação	19
2.1.2	Aprendizado não supervisionado	19
2.1.3	Aprendizado por reforço	19
2.2	Modelos	20
2.2.1	CrITÉrios de performance	20
2.2.2	Modelos e classificadores lineares	22
2.2.2.1	Regressão linear	22
2.2.2.2	Perceptron	23
3	<i>DEEP LEARNING</i>	26
3.1	Redes neurais multicamadas	26
3.1.1	Retropropagação	27
3.2	Redes neurais recorrentes	29
3.2.1	Treinamento de RNNs	30
3.3	<i>Reservoir computing</i>	31
4	<i>RESERVOIR COMPUTING FOTÔNICO</i>	35
4.1	ReservatÓrios fotônicos integrados em paralelo	36
4.2	ReservatÓrios multiplexados no tempo	36
5	MICRORRESSONADORES EM ANEL	39
5.1	<i>All-pass</i>	40
5.2	<i>Add-drop</i>	41
5.3	Parâmetros de desempenho	42
6	DESENVOLVIMENTO METODOLÓGICO	44
6.1	Photontorch	44
6.1.1	Formalismo tensorial	44

6.1.2	Esquemático e equacionamento do circuito	47
6.2	Caracterização do ressonador em anel e análise comportamental . .	49
6.2.1	Parâmetros de design	49
6.2.2	Resposta em frequência	50
6.2.3	Resposta ao degrau	51
6.3	Aplicação do circuito em RC	52
6.3.1	Performance	54
6.3.2	Resposta dinâmica do reservatório	57
6.3.3	Performance em função do comprimento de onda	58
6.4	Impacto da variação de temperatura	60
6.4.1	Condições de acoplamento	63
7	DISCUSSÃO DE RESULTADOS	65
8	CONCLUSÃO	66
	REFERÊNCIAS	68

1 INTRODUÇÃO

Atualmente, o processamento de informações digitais é primordial para o funcionamento da sociedade moderna. Nesse contexto, o campo de *machine learning* (aprendizado de máquina), no centro das ciências de dados e da Inteligência Artificial (IA), tem se tornado uma ferramenta essencial para a transformação digital das empresas e para o cenário de *Big Data* [1].

Na interface entre a análise estatística e a informática, o domínio de *machine learning* se encarrega da modelização de dados. Os princípios elementares da área tiveram origem em temas de outras esferas, como a inferência frequentista e bayesiana, IA e tratamento de sinais, abrangendo diferentes métodos de análise de dados. Entretanto, o campo vem sendo impulsionado, principalmente, pelo desenvolvimento emergente de modelos e técnicas avançadas em redes neurais artificiais (RNA) e *deep learning*. Essas tecnologias buscam simular a forma como o cérebro humano processa informações, permitindo que sistemas computacionais aprendam, interpretem dados complexos e tomem decisões de maneira autônoma por meio de algoritmos [2]. No entanto, à medida que os modelos de IA se tornam maiores e exigem maior capacidade computacional, a computação tradicional enfrenta desafios significativos para atender às demandas crescentes do mercado.

De acordo com a Lei de Moore, desde a popularização do transistor na década de 1960, o número desses componentes em um microprocessador e, conseqüentemente, o poder computacional dos chips, deveria dobrar a cada 18 meses, aproximadamente. Essa previsão estabeleceu-se como uma regra prática na indústria de semicondutores e tornou-se um dos principais impulsionadores dos rápidos avanços tecnológicos das últimas décadas. Entretanto, com o advento da Internet e o surgimento de tecnologias mais sofisticadas, a necessidade de miniaturização, melhor desempenho e maior largura de banda têm crescido expressivamente, enquanto a Lei de Moore começa a estagnar [3], como ilustrado na Fig. 1.

Para acomodar a crescente densidade de elementos em microchips cada vez menores, o tamanho dos transistores já atinge a escala atômica, o que introduz limitações de natureza quântica. Ao reduzir suas dimensões, o valor binário torna-se instável devido a efeitos térmicos, além de gerar desafios de natureza econômica, já que os custos de produção e pesquisa aumentam consideravelmente [5].

Nesse mesmo cenário, seguindo uma previsão conservadora, a carga computacional para modelos de IA está projetada para crescer mil vezes até 2030 [6]. Para atender a essa crescente demanda, soluções como as unidades de processamento gráfico (GPUs) e as unidades de processamento de tensores da Google (TPUs), que se beneficiam da enorme capacidade de paralelização das RNAs, têm sido amplamente utilizadas para realizar o

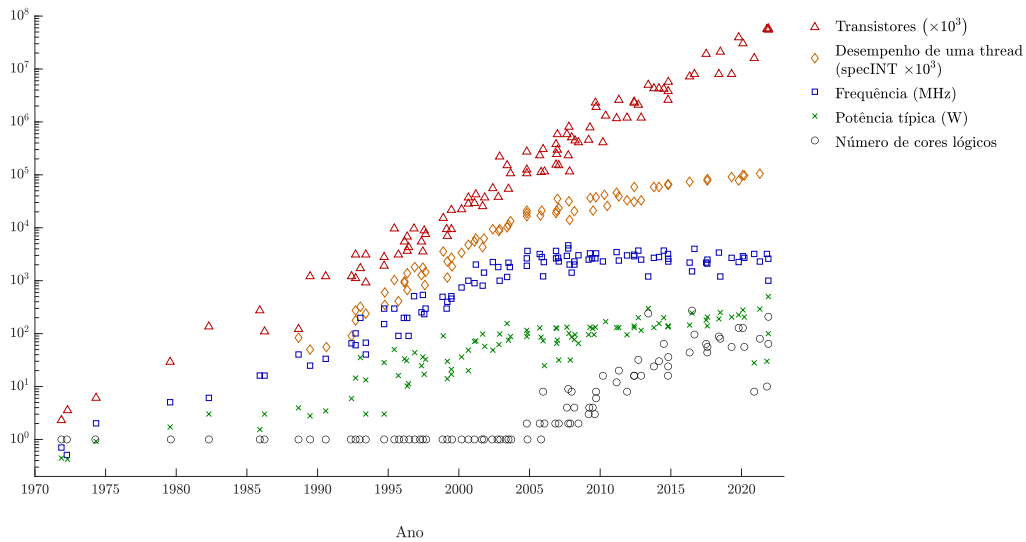


Figura 1 – Curvas de tendência a partir de dados sobre microprocessadores. [4]

processamento paralelo de dados. Embora, no momento, esses recursos ofereçam uma certa aceleração de desempenho com potencial para mais alguns anos, eles permanecem limitados pelos mesmos problemas enfrentados pela Lei de Moore e, com o tempo, chegarão a um limite [7]. Além disso, o crescente consumo de energia associado às aplicações de IA representa uma preocupação significativa, tanto do ponto de vista econômico quanto ambiental, o que reforça os desafios no desenvolvimento dos processadores das novas gerações para atender às perspectivas presentes e futuras desses serviços [8].

Buscando, então, reduzir a disparidade entre o poder computacional e as exigências crescentes, novas pesquisas têm explorado tecnologias alternativas para oferecer um esquema de processamento de dados mais rápido, inteligente e eficiente [9]. Nesse contexto, inspirada em paradigmas de IA e, conseqüentemente, na estrutura e no funcionamento do cérebro humano, surge a computação neuromórfica, que busca, como extensão desse conceito, desenvolver arquiteturas alternativas [10] capazes de processar grandes volumes de dados de forma mais eficiente, com menor consumo de energia em comparação aos processadores convencionais.

No lugar dos transistores utilizados na computação tradicional, a computação neuromórfica explora novos materiais e sistemas para simular redes de neurônios artificiais e sinapses que interagem de maneira fundamentada nos processos cognitivos humanos. Enquanto os computadores convencionais apresentam bom desempenho para executar rotinas sequenciais, sua ineficiência se destaca no processamento distribuído e paralelo, essencial para aplicações de RNA. A abordagem neuromórfica é, portanto, vista como uma solução promissora para superar as limitações da arquitetura de von Neumann (processamento centralizado e memória separada) e consolidar novos tipos de computadores capazes de realizar atividades para as quais os humanos são altamente aptos.

Para além de sistemas eletrônicos e biológicos, uma das formas mais eficazes que o campo da computação neuromórfica encontrou para aumentar a velocidade e a eficiência das operações foi explorar a luz como portadora de informações, método já amplamente utilizado e consolidado nas telecomunicações. Os computadores fotônicos usufruem de várias propriedades do domínio óptico. Nesse panorama, a alta velocidade de propagação da luz e as não linearidades em óptica ultrarrápida de alguns materiais podem ser utilizadas para assegurar um processamento de dados mais rápido em comparação às tecnologias tradicionais baseadas em elétrons. Ademais, os fótons permitem transmitir grandes volumes de informações em paralelo, com baixo consumo de energia e mínima dissipação térmica.

Para fins de comparação, no domínio eletrônico, sinais digitais começam a apresentar problemas significativos de processamento a partir de 10 Gbps devido ao efeito pelicular e às características indutivas, enquanto transmissões de dados ópticas facilmente atingem 100 Gbps com menores distorções de sinal [11]. Adicionalmente, ondas eletromagnéticas apresentam amplitude e fase, que conferem dois graus de liberdade a serem explorados para melhorar as computações, além de uma largura de banda disponível muito maior do que a utilizada por chips eletrônicos.

1.1 Fotônica em silício

Dedicada ao estudo da interação da luz com a matéria em diferentes meios, a fotônica aprofunda-se na aplicação de fenômenos ópticos para a transmissão e o processamento de dados, a partir da geração, propagação e detecção da luz, geralmente no domínio visível (380-780 nm) e no infravermelho próximo (780-2000 nm). Os avanços na fotônica têm sido fundamentais para diversas tecnologias, incluindo comunicações ópticas, sensores, iluminação e geração fotovoltaica.

Nas últimas décadas, grupos de pesquisa e empresas têm concentrado extensivamente seus esforços na inclusão de circuitos ópticos miniaturizados em circuitos integrados, permitindo a incorporação de uma ampla gama de componentes em um único chip fotônico. Nessa conjuntura, a plataforma de integração que se consolidou como a mais promissora foi a fotônica em silício, possibilitando o desenvolvimento de processadores e transceptores ópticos de custo reduzido e excelente desempenho para aplicações em telecomunicações e processamento de dados [12].

Objetivando criar estruturas apropriadas para propagação em um chip de silício em isolante (*silicon-on-insulator*, SOI), canais de silício ($n = 3,47$ em 1550 nm) são usualmente gravados em um substrato laminar de óxido de silício (SiO_2 , $n = 1,44$ em 1550 nm) e revestidos por uma camada de baixo índice refrativo (ar ou óxido). Nesses canais, devido ao grande contraste de índice de refração entre o núcleo de silício e o revestimento, a luz permanece fortemente confinada lateral e transversalmente, dando origem a guias de onda. Verificou-se que as propriedades do guia, incluindo a condição de propagação monomodo,

estão intimamente vinculadas às dimensões da seção transversal da estrutura, que devem estar na ordem submicrométrica. Os valores mais utilizados na fabricação situam-se entre 400 e 500 nm de largura e 200 a 300 nm de altura para guias de alto contraste [13,14].

A fotônica em silício se beneficiou significativamente da compatibilidade com semicondutores de óxido de metal complementar (CMOS), cuja plataforma de integração já se encontra bem desenvolvida e oferece ferramentas e processos de fabricação em massa e de baixo custo. Nesse contexto, usando guias de silício como elementos básicos, é possível implementar uma variedade de componentes ópticos, como acopladores direcionais [15], redes de Bragg [16], interferômetros de Mach-Zehnder [17], ressoadores em anel [18] e moduladores ópticos [19].

Apesar de a fotônica em silício ser atualmente considerada uma plataforma tecnológica madura, o desenvolvimento na área ainda enfrenta desafios, como a custosa incompatibilidade de acoplamento dos chips fotônicos com componentes em fibra óptica, devido à grande disparidade dimensional entre os modos de propagação desses e os dos guias de onda em silício [20]. Contudo, essas adversidades têm sido pouco a pouco solucionadas com os recentes avanços. É o caso, por exemplo, da integração de fontes nanolaser na plataforma: uma vez que o silício apresenta ineficiente radiação por recombinação, estruturas alternativas com semicondutores III-V, como o arseneto de gálio (GaAs), têm sido propostas para garantir a emissão laser integrada [21].

1.2 Computação neuromórfica e *reservoir computing* fotônicos

Apoiada pelas promessas de propagação eficiente de dados em alta velocidade e ampla largura de banda, proporcionadas pelos avanços da fotônica em silício, a computação neuromórfica já reúne uma gama de implementações ópticas em hardware [22], que evidenciam as melhorias propostas para operações analógicas. Esses avanços, entretanto, não devem ser interpretados como uma solução universal para a aceleração de hardware, capaz de substituir completamente os modelos digitais, visto que a maioria das computações demanda uma interface eletrônica. Nesse sentido, a computação neuromórfica fotônica é compreendida como uma alternativa promissora para realizar processamentos complexos em tempo real, nos quais hardwares puramente eletrônicos não conseguem oferecer o melhor desempenho.

Para isso, os hardwares fotônicos utilizam os elementos fundamentais das RNAs: unidades não lineares (neurônios) interconectadas por pesos lineares (sinapses) e um sistema de representação de dados. Esse esquema permite a reprodução de modelos de aprendizagem profunda [23], modelos convolucionais [24], modelos de disparos neuronais [25] e computadores de reservatório.

Por sua vez, a computação por reservatório, do inglês *reservoir computing*, tem

sido uma das áreas de crescente destaque no campo de sistemas neuromórficos [26] devido à sua versatilidade. Seu mecanismo baseia-se em uma rede de neurônios não lineares, aleatoriamente conectados (reservatório), que, a partir de um determinado sinal de entrada em tempo real, gera um número extenso de estados de resposta, cuja combinação linear conduz à solução de alguma operação temporal ou sequencial desejada.

Muitos substratos físicos podem ser utilizados como reservatório, incluindo sistemas eletrônicos, mecânicos e biológicos [27]. No campo da fotônica, é possível encontrar reservatórios propostos a partir de diferentes arquiteturas integradas em chips.

Recentemente, uma arquitetura composta, em seu cerne, por um único microrressonador em anel mostrou-se adequada para o funcionamento de um reservatório [28–30]. Os regimes linear e não linear do microrressonador foram explorados, em modelos teóricos e práticos, como forma de realizar inferências e operações booleanas, demonstrando excelente competência computacional e capacidade de memória. A topologia analisada não se baseia em uma distribuição espacial significativa de núcleos e componentes ópticos, o que permite reduzir consideravelmente a área ocupada em chips integrados, especialmente no caso de reservatórios mais complexos com um número elevado de neurônios.

1.3 Objetivos

Apesar das variadas provas de conceito voltadas para a validação de novas arquiteturas como reservatórios, poucas análises foram realizadas com relação ao impacto da temperatura na performance de chips integrados de computadores de reservatório fotônicos.

Flutuações de temperatura em componentes ópticos provocam mudanças no índice de refração e, conseqüentemente, flutuações de fase, no que se refere ao coeficiente termo-óptico dn/dT dos materiais. Na metrologia, essa propriedade já foi utilizada para produzir, com sucesso, sensores de temperatura compactos e de fácil integração em chips [31–33]. Entretanto, o ruído termorrefrativo pode ter impacto significativo no desempenho de computadores neuromórficos baseados em substratos fotônicos, uma vez que a dinâmica dos sistemas físicos é alterada.

Este trabalho tem como objetivo principal analisar, por meio de simulações, o efeito da variação de temperatura na robustez de um sistema de *reservoir computing* composto por um microrressonador em anel, cujo *design* será proposto. Inicialmente, é realizado um estudo comportamental e descritivo do microrressonador, permitindo uma compreensão aprofundada da dinâmica de funcionamento do sistema implementado. Posteriormente, antes de abordar a análise térmica, é apresentada uma prova de conceito demonstrando que o sistema é capaz de operar como um reservatório. Para tal, utiliza-se a operação XOR como um problema de *machine learning* a ser resolvido pelo computador.

Este trabalho inicia com uma revisão teórica dos principais eixos temáticos rela-

cionados ao tema. Nos capítulos 2 e 3, são apresentados os conceitos fundamentais do campo de *machine learning*. No Capítulo 4, é descrito o estado atual do conhecimento e das tecnologias no domínio da computação neuromórfica fotônica. Por fim, o Capítulo 5 aborda a teoria que embasa o funcionamento dos microrressonadores em anel.

Nos capítulos subsequentes, são apresentados os procedimentos metodológicos, seguidos pelos resultados obtidos em simulação e pela discussão dos dados relacionados ao objetivo principal do projeto.

2 FUNDAMENTOS DE MACHINE LEARNING

Na ciência da computação, as aplicações tradicionais utilizam procedimentos bem definidos e entradas de dados para gerar respostas específicas. No entanto, quando os programas são capazes de aprender sem que suas instruções tenham sido previamente programadas para esse propósito, estamos no campo do aprendizado de máquina, ou *machine learning*.

Dessa forma, pode-se entender um programa de aprendizado automático como aquele capaz de desempenhar tarefas ou modificar seu comportamento com base na experiência. Em termos práticos, isso significa utilizar dados e respostas para estabelecer um mecanismo que permita prever as respostas a partir dos dados. Nesse contexto, uma coleção organizada e suficientemente abundante de dados, a partir da qual ocorre o processo de aprendizado, constitui um dos pilares fundamentais para os programas de *machine learning*, juntamente com o algoritmo de aprendizado e o modelo gerado.

Quando técnicas de *machine learning* são empregadas, frequentemente busca-se transformar os dados de alguma forma. Um exemplo disso é a conversão das leituras de sensores de um equipamento em uma classificação de comportamento normal ou irregular. Alternativamente, pode-se desejar compreender a relação entre o aumento de vendas e as estratégias de propaganda adotadas anteriormente por uma empresa. Denomina-se modelo o sistema computacional que recebe os dados, realiza a transformação e produz uma predição ou inferência.

Para ajustar os parâmetros dos modelos e otimizá-los para seus objetivos específicos, os dados são processados por algoritmos de aprendizado em uma etapa chamada treinamento. Após esse processo, os modelos de *machine learning* tornam-se valiosos para a resolução de problemas de natureza desconhecida, para casos que são difíceis de formalizar em termos algorítmicos tradicionais ou para oferecer soluções mais eficientes, tanto em custo quanto em complexidade computacional.

O campo de *machine learning* é bastante vasto e aborda uma série de tipos de problemas.

2.1 Tipos de problemas em *machine learning*

2.1.1 Aprendizado supervisionado

Uma das principais classes de problemas em *machine learning* é o aprendizado supervisionado, cujo objetivo é realizar predições a partir de um conjunto de informações rotuladas, isto é, dados acompanhados dos resultados esperados. Nesse contexto, sejam dadas n amostras $\{\vec{x}_i\}_{i=1,\dots,n}$, onde cada amostra \vec{x}_i é um elemento do espaço amostral \mathcal{X} ,

e seus respectivos rótulos $\{y_i\}_{i=1,\dots,n}$, onde cada rótulo y_i pertence ao espaço \mathcal{Y} . Supõe-se que os rótulos podem ser obtidos a partir das amostras por meio de uma função $\phi : \mathcal{X} \rightarrow \mathcal{Y}$, que é fixa e desconhecida.

$$y_i = \phi(\vec{x}_i) + \epsilon_i, \quad (2.1)$$

onde ϵ_i é um erro aleatório. Assim, o objetivo formal do aprendizado supervisionado é utilizar os dados para determinar uma função de decisão $f : \mathcal{X} \rightarrow \mathcal{Y}$, tal que, para todo par $(\vec{x}, \phi(\vec{x})) \in \mathcal{X} \times \mathcal{Y}$, seja válido que $f(\vec{x}) \approx \phi(\vec{x})$. O conjunto $\mathcal{D} = \{(\vec{x}_i, y_i)\}_{i=1,\dots,n}$ representa, então, o conjunto de dados.

Por um lado, o espaço no qual estão definidas as amostras é, geralmente, $\mathcal{X} = \mathbb{R}^p$. Em termos práticos, cada amostra é descrita por p atributos, de modo que a matriz $X \in \mathbb{R}^{n \times p}$, onde $X_{ij} = x_i^j \in \mathbb{R}$ representa o j -ésimo atributo da i -ésima amostra, é conhecida como a matriz de dados. Os atributos podem ser entendidos como informações características das amostras: em uma análise de clima, por exemplo, as amostras medidas podem conter atributos de temperatura, pressão, umidade do ar, data da medição, localidade, etc.

Por outro lado, o espaço no qual os rótulos estão definidos pode variar, determinando tipos específicos de problemas: (i) no caso de $\mathcal{Y} = \mathbb{R}$, trata-se de um problema de regressão; (ii) no caso de rótulos binários, $\mathcal{Y} = 0, 1$ ou $-1, 1$, tem-se um problema de classificação binária; (iii) no caso de $\mathcal{Y} = 1, 2, 3, \dots, C$, tem-se um problema de classificação multiclasse.

2.1.1.1 Regressão

A regressão é uma das tarefas mais tradicionais e simples dentro do aprendizado supervisionado. Sua premissa parte do princípio de que a forma analítica da função de decisão é supostamente conhecida, configurando-se, particularmente, como uma ferramenta bastante útil para prever respostas quantitativas [34]. Seja a função de forma linear, polinomial ou logística, os problemas de regressão são usualmente solucionados por meio de um modelo paramétrico. Basicamente, dado um conjunto $\mathcal{D} = \{(\vec{x}_i, y_i)\}_{i=1,\dots,n}$, composto por n amostras de dimensão p e seus respectivos rótulos reais, os algoritmos de aprendizado em regressão buscam otimizar os parâmetros $\vec{w} \in \mathbb{R}^m$ de um modelo que é função dos atributos das amostras. De maneira geral, o valor esperado é estimado por

$$\hat{y} = f(\vec{x} | \vec{w}) \quad (2.2)$$

A complexidade de modelos paramétricos aumenta à medida que se incrementa o número de parâmetros a serem aprendidos ou otimizados, ou seja, conforme o número de atributos utilizados cresce. Em contraposição, os modelos não paramétricos tendem a aumentar sua complexidade com o número de amostras no conjunto de dados.

2.1.1.2 Classificação

Enquanto a regressão se mostra apropriada para respostas quantitativas, muitas situações em *machine learning* demandam soluções qualitativas e categóricas, para as quais os modelos descritos anteriormente podem não ser os mais adequados. Nesse contexto, o processo de determinar qualitativamente a classe de uma amostra é denominado classificação.

Na classificação binária, os modelos preditivos podem assumir diretamente a forma de uma função f com valores em 0, 1 ou, alternativamente, utilizar uma função intermediária $g : \mathcal{X} \rightarrow \mathbb{R}$, que atribui a uma amostra um valor proporcional à sua suscetibilidade de pertencer à classe 1. Esse valor pode ser, por exemplo, a probabilidade de a amostra pertencer a tal classe [35]. Nesse contexto, f é obtida aplicando um limiar à função de decisão g .

O conceito de função de decisão, então, permite particionar o espaço amostral \mathcal{X} em duas regiões de decisão, \mathcal{R}_0 e \mathcal{R}_1 , tais que $\mathcal{R}_0 = \{\vec{x} \in \mathcal{X} \mid g(\vec{x}) \leq 0\}$ e $\mathcal{R}_1 = \{\vec{x} \in \mathcal{X} \mid g(\vec{x}) > 0\}$. Essa mesma definição pode ser estendida de forma geral para o caso de classificação multiclasse. Nesse caso, haverá C fronteiras de decisão no espaço amostral para um problema multiclasse de C classes.

2.1.2 Aprendizado não supervisionado

Ao contrário do aprendizado supervisionado, no aprendizado não supervisionado o conjunto de dados de entrada não é rotulado. Assim, sejam dadas n amostras $\{\vec{x}_i\}_{i=1,\dots,n}$ descritas em um espaço \mathcal{X} , cada uma com p atributos, o aprendizado não supervisionado não tem como objetivo a determinação de um modelo preditivo. Em vez disso, busca-se explorar propriedades interessantes dentro do conjunto de dados. Entre esses objetivos, destaca-se o desenvolvimento de métodos informativos de visualização, que permitem identificar padrões nos dados, possibilitando a determinação de subgrupos entre os atributos ou as amostras.

2.1.3 Aprendizado por reforço

No contexto do aprendizado por reforço, o sistema de aprendizagem é livre para interagir com o ambiente e tomar decisões. A cada ação, ou sequência de ações, o sistema recebe uma recompensa, que pode ser positiva, se a decisão for favorável, ou negativa, em caso contrário. O processo de aprendizado consiste, portanto, em definir uma estratégia que permita à máquina obter sistematicamente a melhor recompensa possível. As principais aplicações do aprendizado por reforço concentram-se em áreas como jogos e robótica.

Pela natureza do projeto abordado, a revisão teórica deste trabalho concentra-se, sobretudo, nas particularidades essenciais do aprendizado supervisionado.

2.2 Modelos

2.2.1 Critérios de performance

Como discutido na Seção 2.1.1, seja \mathcal{F} o espaço de modelos possíveis, a resolução de um problema de aprendizado supervisionado envolve a obtenção de uma função $f \in \mathcal{F}$, cujas previsões estejam o mais próximas possível dos valores rotulados das amostras. A qualidade do modelo escolhido pode, portanto, ser quantificada por meio de funções de custo $L : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}$. Essas funções de custo, também conhecidas como funções de perda, $L(y, f(\vec{x}))$, mapeiam a distância entre os resultados reais e os valores preditos.

Nessas condições, busca-se o modelo cuja função f minimize a esperança do erro (custo) no conjunto de valores possíveis de $\vec{x} \in \mathcal{X}$. De maneira formal, considerando o conjunto de dados $\{(\vec{x}_i, y_i)\}_{i=1, \dots, n}$, tem-se:

$$f = \arg \min_{h \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^n L(h(\vec{x}_i), y_i) \quad (2.3)$$

Existem diferentes possibilidades de funções de custo para as distintas categorias de problemas de aprendizado supervisionado. A escolha de uma função de custo depende, portanto, tanto de sua relevância na aplicação prática quanto de sua complexidade associada ao processo de otimização.

Quando os algoritmos de aprendizagem produzem modelos a partir de um conjunto de dados rotulados de treinamento, considera-se como parâmetro fundamental o erro mínimo entre os rótulos e os valores preditos. No entanto, o desempenho de um modelo não se limita à etapa de treinamento, pois o objetivo é que ele seja igualmente capaz de fazer previsões corretas para dados de entrada que não tenham sido previamente utilizados em sua construção. Em outras palavras, um bom modelo de *machine learning* é aquele que consegue generalizar.

Dessa forma, por um lado, um modelo não deve ser tão simples a ponto de apresentar um desempenho insatisfatório até mesmo no conjunto de dados de treinamento. Por outro lado, também não deve ser tão complexo a ponto de se ajustar excessivamente aos dados de treinamento, capturando também o ruído associado e fornecendo previsões imprecisas para novos dados, pois se torna “especializado” nas amostras de treino. Na Fig. 2, são ilustradas, respectivamente, essas duas situações de sub e sobreaprendizado.

Para mitigar o risco de ajuste excessivo, é desejável limitar a complexidade do modelo, o que tradicionalmente é feito pela redução do número de parâmetros utilizados. No entanto, uma abordagem mais geral baseia-se na regularização dos parâmetros do modelo [36]. Nesse contexto, a regularização consiste em adicionar um termo, proporcional à complexidade do modelo, na função de custo, penalizando, geralmente, modelos com parâmetros de valores elevados. Assim, a formalização da Eq. 2.3 incorpora um termo de

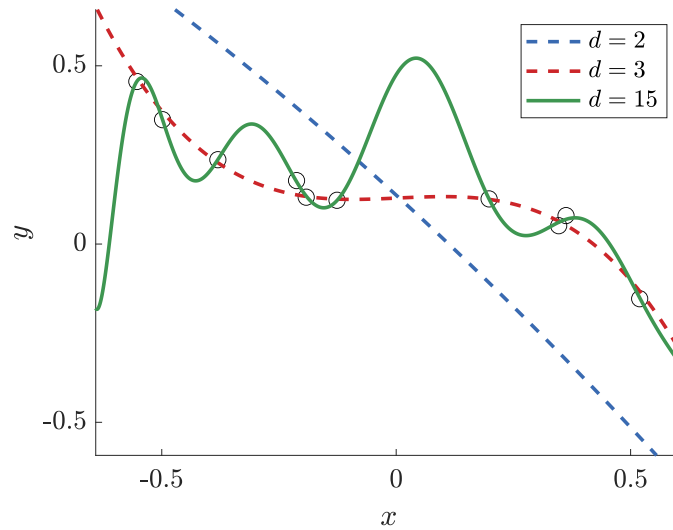


Figura 2 – Os rótulos y das amostras (representadas por pontos) foram produzidos a partir de um polinômio de ordem $d = 3$. O modelo de grau $d = 2$ é uma péssima aproximação dos dados, indicando subaprendizado. O modelo de grau $d = 15$, cujo erro/custo é mais baixo, se ajusta excessivamente aos dados, indicando sobreaprendizado.

restrição Ω sobre os modelos possíveis:

$$f = \arg \min_{h \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^n L(h(\vec{x}_i), y_i) + \lambda_{reg} \Omega(h) \quad (2.4)$$

onde o coeficiente de regularização $\lambda_{reg} \in \mathbb{R}_+$ controla a importância relativa de cada um dos termos.

De modo geral, o erro da função de custo associado apenas aos dados que foram utilizados para construir o modelo é um péssimo estimador do erro de generalização, isto é, do erro do modelo em relação ao conjunto de todos os dados possíveis. Se um modelo estiver excessivamente ajustado aos dados de treinamento, o valor da função de custo será praticamente nulo, enquanto o erro de generalização pode ser arbitrariamente grande. Portanto, para avaliar adequadamente um modelo, é indispensável a utilização de um conjunto de dados rotulados de teste, que não tenha sido previamente utilizado pelo algoritmo de aprendizagem.

A compreensão da relação entre a complexidade do modelo, a escolha de funções de custo e a regularização é fundamental para o desenvolvimento de algoritmos de aprendizado supervisionado eficazes. A capacidade de um modelo em generalizar para novos dados depende não apenas de um ajuste apropriado aos dados de treinamento, mas também da utilização de técnicas que garantam um desempenho consistente em contextos práticos. Nesse sentido, modelos lineares emergem como uma abordagem essencial, oferecendo uma combinação de simplicidade e eficácia, além de serem amplamente utilizados em diversas

aplicações de aprendizado.

2.2.2 Modelos e classificadores lineares

2.2.2.1 Regressão linear

A presunção de linearidade em um modelo significa que o valor esperado $f(\vec{x})$ é expresso como uma combinação linear, ou seja, uma soma ponderada dos atributos de \vec{x} . Na regressão linear, a função de decisão f assume a forma (Eq. 2.2):

$$f : \vec{x} \rightarrow w_0 + \sum_{j=1}^p w_j x_j \quad (2.5)$$

onde os elementos de $\vec{w} \in \mathbb{R}^{p+1}$ são denominados pesos da regressão e configuram os coeficientes ou parâmetros do modelo. A Eq. 2.5 pode ser generalizada na forma matricial:

$$\hat{Y} = w^T X \implies [\hat{y}_1 \quad \dots \quad \hat{y}_n] = [w_0 \quad w_1 \quad \dots \quad w_p] \begin{bmatrix} 1 & \dots & 1 \\ x_1^1 & \dots & x_n^1 \\ \vdots & \ddots & \vdots \\ x_1^p & \dots & x_n^p \end{bmatrix} \quad (2.6)$$

Na prática, os valores de cada peso são inicialmente desconhecidos. Os dados de treinamento $\{(\vec{x}_i, y_i)\}_{i=1, \dots, n}$ devem ser utilizados para estimá-los de forma que os valores esperados \hat{y}_i do vetor \hat{Y} sejam o mais próximos possíveis dos valores reais rotulado y_i , como discutido na Seção 2.2.1. No caso da regressão linear, a técnica mais utilizada para essa etapa é o método dos mínimos quadrados, que busca minimizar a função de erro quadrático $L(\vec{w})$ dada por:

$$L_i(\vec{w}) = \frac{1}{2} (y_i - \hat{y}_i)^2 \quad (2.7)$$

Durante o treinamento, busca-se parâmetros \vec{w}^* que minimizem o custo total em todas as amostras de treinamento:

$$\vec{w}^* = \arg \min_{\vec{w} \in \mathbb{R}^{p+1}} \frac{1}{n} \sum_{i=1}^n L_i(\vec{w}) \quad (2.8)$$

É possível demonstrar que, calculando o gradiente da Eq. 2.7 com relação a \vec{w} e anulando-o [37], obtém-se

$$\vec{w}^* = (X X^T)^{-1} X \vec{y}^T \quad (2.9)$$

2.2.2.2 Perceptron

Os primeiros modelos neuronais surgiram de esforços para uma melhor compreensão do cérebro humano. O perceptron, como ficou denominado o primeiro modelo neuronal implementado em hardware [38], foi concebido com o objetivo de modelizar matematicamente o tratamento de informações pela rede de neurônios biológica, encontrada no córtex dos mamíferos.

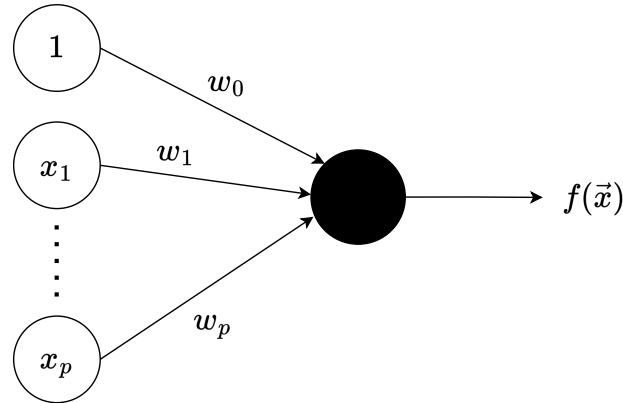


Figura 3 – Arquitetura de um perceptron

Em sua essência, o perceptron configura um esquema semelhante ao da regressão. Como visto na Fig. 3, o modelo é composto por uma camada de entrada formada por $p + 1$ neurônios, ou unidades, que correspondem cada um a um dos atributos de uma amostra. Essas unidades transmitem então seus valores de entrada para a camada seguinte, formada por um único neurônio (em preto). Esta unidade de saída, primeiramente, computa uma combinação linear $o(\vec{x}) = w_0 + \sum_{j=1}^p w_j x_j$ dos sinais recebidos $\vec{x}_i = (1, x_1, x_2, \dots, x_p)$ em sua entrada e, em seguida, aplica uma função de ativação a , cujo resultado será a saída do perceptron:

$$f(\vec{x}) = a(o(\vec{x})) = a\left(w_0 + \sum_{j=1}^p w_j x_j\right) \quad (2.10)$$

Nesse âmbito, os termos w_j são denominados pesos de conexão entre a unidade de entrada j e o neurônio de saída.

A função de ativação desempenha um papel fundamental no modelo do perceptron, especialmente ao lidar com problemas de *machine learning* mais complexos. Em casos de regressão linear, a função de ativação pode ser tão simplesmente a função identidade. No entanto, em cenários onde a relação entre os atributos de entrada e a resposta não é linear, o uso de funções de ativação não lineares se torna crucial. Essas funções permitem ao perceptron capturar padrões mais sofisticados, tornando possível modelar relações complexas nos dados para as quais os modelos lineares seriam limitados. Assim, o perceptron se torna eficaz tanto em problemas de regressão quanto em problemas de classificação [39].

As primeiras ideias de perceptron utilizavam uma lógica limiar como função de ativação, interpretando os resultados na saída a partir de um discriminante. Esses modelos se mostraram excelentes classificadores binários, capazes de executar certas funções lógicas, como operações do tipo AND e OR [40], cujas tabelas verdades são, respectivamente, apresentadas nas Tab. 1a e Tab. 1b.

x_1	x_2	$x_1 \wedge x_2$
0	0	0
0	1	0
1	0	0
1	1	1

(a) AND

x_1	x_2	$x_1 \vee x_2$
0	0	0
0	1	1
1	0	1
1	1	1

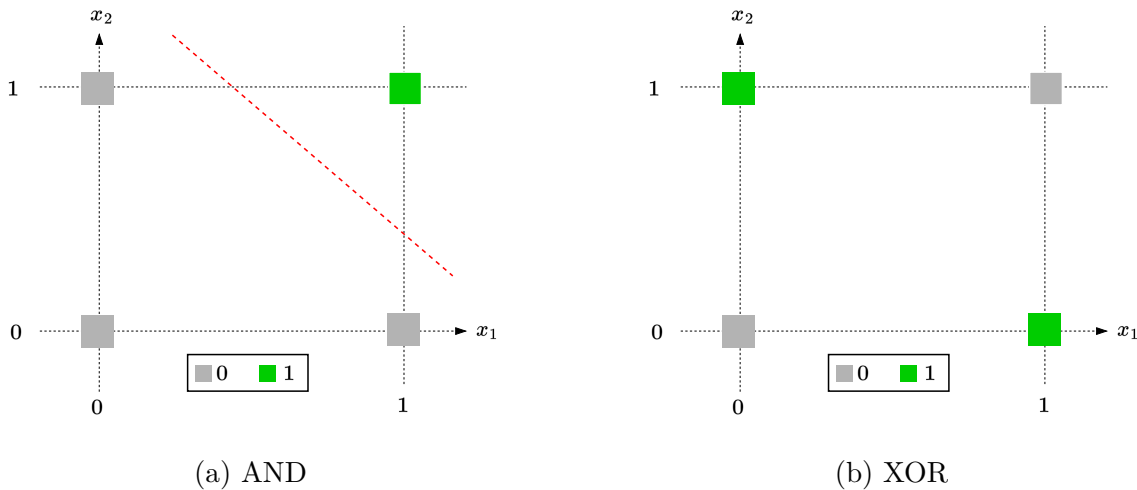
(b) OR

x_1	x_2	$x_1 \oplus x_2$
0	0	0
0	1	1
1	0	1
1	1	0

(c) XOR

Tabela 1 – Tabelas verdades de operações lógicas

Como discutido na Seção 2.1.1.2, o objetivo principal da classificação é dividir o espaço amostral em regiões de decisão. Tomando como exemplo a operação lógica AND, representada bidimensionalmente na Fig. 4a, observa-se que os dados são linearmente separáveis. Em outras palavras, uma simples função linear (em vermelho) é capaz de definir uma fronteira de decisão entre os dois possível resultados, resolvendo o problema de forma eficaz. Nesse contexto, espera-se que modelos e algoritmos de treinamento lineares, como o perceptron, apresentem, de fato, um desempenho excelente.



(a) AND

(b) XOR

Figura 4 – Operações lógicas mapeadas bidimensionalmente. Em (a) é ilustrada a possibilidade de separação linear

Entretanto, no caso da operação lógica XOR, cuja tabela verdade é apresentada na Tab.1c, a situação se torna mais complexa. Como mostrado na Fig.4b, os resultados binários são mapeados de forma cruzada. Nesse cenário, não é possível particionar os dados por meio de uma reta. Assim, o modelo do perceptron, limitado por sua natureza linear, torna-se instável ao tentar resolver o problema dessa função lógica e de outros tipos de classificação semelhantes.

Após o entusiasmo inicial gerado pelos primeiros modelos conexionistas, as limitações observadas no perceptron impulsionaram o desenvolvimento de modelos mais robustos. Nesse contexto, surgiram as primeiras arquiteturas de redes neurais multicamadas, que serviram como base para o desenvolvimento de modelos mais complexos no campo do *deep learning* [41].

3 DEEP LEARNING

O desenvolvimento das técnicas de *deep learning* revolucionou o processamento de grandes volumes de dados e a resolução de problemas complexos em áreas como visão computacional, processamento de linguagem natural e previsão de séries temporais.

A área representa uma vertente bastante ativa de *machine learning* e das ciências de dados, responsável pelo estudo das redes neurais profundas, compostas por duas ou mais camadas de neurônios entre a camada de entrada e a camada de saída do perceptron. Entre as principais arquiteturas do campo estão as redes neurais multicamadas, as redes neurais recorrentes e as redes neurais convolucionais. Neste capítulo, abordam-se os fundamentos das duas primeiras, com ênfase em uma técnica mais moderna de redes neurais recorrentes: o *reservoir computing*.

3.1 Redes neurais multicamadas

Nas arquiteturas de redes neurais multicamadas, ou perceptron multicamadas (MLP), o modelo inicial do perceptron recebe as denominadas camadas ocultas de neurônios (*hidden layers*) entre a camada de entrada e a de saída. Como apresentado na Fig. 5, nessa topologia, cada neurônio de uma das camadas intermediárias, ou da camada de saída, recebe, como entradas, as saídas de neurônios da camada antecedente.

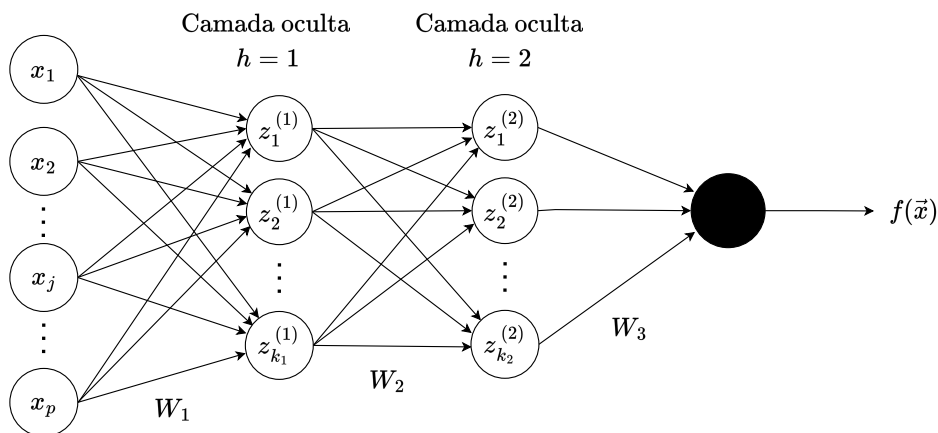


Figura 5 – Arquitetura ilustrativa de uma rede neural com duas camadas ocultas. As notações W_h representam as matrizes inteiras de pesos $w_{jk}^{(h)}$ entre os neurônios da camada $h - 1$ que alimenta a subseqüente h .

Nesse esquema, sejam $w_{jk}^{(h)}$ os pesos de conexão entre o neurônio j da camada $h - 1$ e o neurônio k da camada h ; a_h a função de ativação na saída da camada h ; e k_h o número de neurônios na camada h , é possível equacionar a saída do k -ésimo neurônio $z_k^{(1)}$ da camada oculta $h = 1$ como:

$$z_k^{(1)} = a_1 \left(\sum_{j=1}^p w_{jk}^{(1)} x_j \right), \quad 1 \leq k \leq q_1 \quad (3.1)$$

De forma análoga, a saída do k -ésimo neurônio $z_k^{(2)}$ da camada oculta $h = 2$ é dada por:

$$z_k^{(2)} = a_2 \left(\sum_{j=1}^{q_1} w_{jk}^{(2)} z_j^{(1)} \right), \quad 1 \leq k \leq q_2 \quad (3.2)$$

Finalmente, o resultado do perceptron da camada de saída resume-se a:

$$f(\vec{x}) = a_3 \left(\sum_{j=1}^{q_2} w_{jk}^{(3)} z_j^{(2)} \right) \quad (3.3)$$

Como não há retorno de dados de uma camada para uma camada precedente, as redes neurais multicamadas são caracterizadas por apresentarem propagação direta (*feedforward*).

Constata-se que a saída do MLP ainda permanece, ao fim, função dos atributos da amostra de entrada, uma vez que a rede neural forma uma cadeia de transformações de \vec{x} capaz de construir modelos complexos percorrendo suas camadas. Dessa forma, ao utilizar funções de ativação não lineares, como a função logística (curva sigmoide), a tangente hiperbólica ou a clássica função ReLU (*Rectified Linear Unit*), definida pela Eq. 3.4, criam-se arquiteturas matemáticas altamente não lineares que permitem resolver um número extenso de problemas de regressão ou classificação.

$$\text{ReLU}(x) = \begin{cases} x, & \forall x > 0 \\ 0, & c.c. \end{cases} \quad (3.4)$$

Teoricamente, uma única camada oculta com um número grande de neurônios poderia aproximar a maior parte das funções $\phi(\vec{x})$. Entretanto, uma boa solução geralmente é encontrada com várias camadas ocultas e uma quantidade razoável de unidades [39]. Nesse sentido, a topologia MLP, que configura trivialmente um modelo paramétrico, apresentará tanto mais parâmetros quanto mais camadas intermediárias e neurônios houver nelas.

O aprendizado desse tipo de arquitetura se faz, portanto, pela determinação dos parâmetros do modelo, ou seja, pelos pesos de conexão entre as unidades, que geralmente são determinados com treinamento por retropropagação (*backpropagation*).

3.1.1 Retropropagação

Seguindo os fundamentos do treinamento de modelos de *machine learning*, a retropropagação consiste em um procedimento de aprendizado que atualiza iterativamente

os pesos das conexões de uma rede neural, com o objetivo de minimizar o erro entre o valor rotulado e o resultado do modelo, representado pela função de custo $L(y_i, f(\vec{x}_i))$ [42]. Nesse método, faz-se fundamentalmente uso do vetor $\nabla_w L$, composto pelos gradientes $\nabla_w L_{jk}^{(h)}$ da função custo em relação aos pesos de conexão da rede. Portanto, para ajustar o parâmetro $w_{jk}^{(h)}$ entre a unidade j da camada $h - 1$ e a unidade k da camada h , deve-se inicialmente computar $\nabla_w L_{jk}^{(h)} = \frac{\partial L(y_i, f(\vec{x}_i))}{\partial w_{jk}^{(h)}}$, que indica a sensibilidade da função de custo ao peso em questão. Em se tratando de funções compostas, pode-se facilitar o cálculo por meio da aplicação da regra da cadeia. Nesse sentido, seja $c_j^{(h)}$ a combinação linear das entradas do neurônio j , tal que $z_j^{(h)} = a_h(c_j^{(h)})$, então:

$$\begin{aligned} \frac{\partial L(y_i, f(\vec{x}_i))}{\partial w_{jk}^{(h)}} &= \frac{\partial L(y_i, f(\vec{x}_i))}{\partial c_k^{(h)}} \frac{\partial c_k^{(h)}}{\partial w_{jk}^{(h)}} = \frac{\partial L(y_i, f(\vec{x}_i))}{\partial z_k^{(h)}} \frac{\partial z_k^{(h)}}{\partial c_k^{(h)}} \frac{\partial c_k^{(h)}}{\partial w_{jk}^{(h)}} \\ &= \left(\sum_{t=0}^{k_{h+1}} \frac{\partial L(y_i, f(\vec{x}_i))}{\partial c_t^{(h+1)}} \frac{\partial c_t^{(h+1)}}{\partial z_k^{(h)}} \right) \frac{\partial z_k^{(h)}}{\partial c_k^{(h)}} \frac{\partial c_k^{(h)}}{\partial w_{jk}^{(h)}} \\ &= \left(\sum_{t=0}^{k_{h+1}} \frac{\partial L(y_i, f(\vec{x}_i))}{\partial c_t^{(h+1)}} w_{kt}^{(h+1)} \right) a_h'(c_k^{(h)}) z_k^{(h-1)} \end{aligned} \quad (3.5)$$

Uma vez que a derivada é determinada, a atualização do peso é realizada efetivamente por meio de uma solução iterativa aproximada, conhecida como algoritmo do gradiente, que explora o fato de que o gradiente $\nabla_w L_{jk}^{(h)}$ indica a direção de maior crescimento da função de custo. De acordo com o método numérico, se um peso $w_{jk}^{(h)}$ é escolhido aleatoriamente e o gradiente é não nulo, então, dado um real positivo η suficientemente pequeno, é possível constatar que $(w_{jk}^{(h)} - \eta \nabla_w L_{jk}^{(h)})$ está mais próximo do mínimo $w_{jk}^{*(h)}$ do que de $w_{jk}^{(h)}$. Dessa forma, o método do gradiente modifica os pesos conforme o algoritmo:

$$w_{jk}^{(h)} \leftarrow w_{jk}^{(h)} - \eta \nabla_w L_{jk}^{(h)} \quad (3.6)$$

até que o valor do gradiente seja menor que uma determinada tolerância. Quanto menor for a tolerância, o peso atualizado estará mais próximo do ponto de mínimo global da função de custo.

O parâmetro de passo η , conhecido como taxa de aprendizado, é essencial no algoritmo de aprendizagem. Um valor muito pequeno de η resulta em convergência lenta, enquanto um valor grande pode causar oscilações em torno do mínimo global ou até a divergência do algoritmo. Embora o método do gradiente não seja o mais rápido em convergência, ele é amplamente utilizado devido à sua simplicidade e à capacidade de ser combinado com técnicas de aceleração, como o Adam [43].

Basicamente, durante o treinamento de uma rede multicamada por retropropagação, para cada amostra de entrada \vec{x}_i e sua saída y_i , ocorre a propagação direta dos dados, armazenando-se os valores das saídas de cada neurônio, obtidas com a solução de equações similares às Eqs. 3.1, 3.2 e 3.3. Posteriormente, na etapa de retropropagação, os erros são usados para atualizar os pesos, começando da camada oculta mais próxima da saída até a camada oculta mais próxima da entrada. Nesse esquema, o desenvolvimento da Eq. 3.5 exemplifica como as derivadas das camadas subsequentes $h + 1$ podem auxiliar no cálculo das derivadas das camadas anteriores h , seguindo o fluxo inverso do método.

3.2 Redes neurais recorrentes

Um dos principais obstáculos das redes neurais de propagação direta é a incapacidade de identificar efetivamente padrões em dados sequenciais, como palavras em uma frase ou informações temporais. Modelos, como o MLP da Seção 3.1, têm dificuldade em capturar relações ao longo de eventos consecutivos.

Problemas dessa categoria levaram ao desenvolvimento de uma nova arquitetura de redes neurais, que explora a retroalimentação, permitindo que unidades de uma camada se conectem com unidades de camadas anteriores. Essa família de topologias é conhecida como redes neurais recorrentes (RNN) [44].

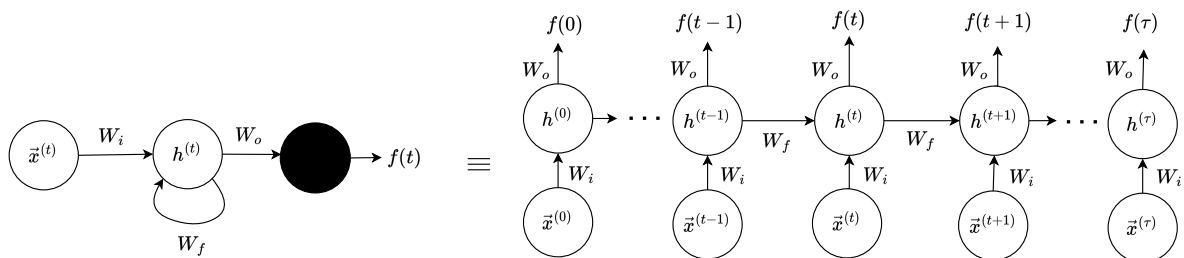


Figura 6 – Esquemático de uma RNN simples. No lado esquerdo, tem-se o diagrama da rede. Do lado direito, tem-se o mesmo circuito expandido, destacando-se a associação das camadas em determinado instante. Os conjuntos de pesos de conexão W_i , W_o e W_f são utilizados a cada processamento sequencial da rede.

O esquema geral de funcionamento de uma RNN pode ser sintetizado na Fig. 6. As entradas do sistema são entendidas como elementos de uma seqüência de amostras, $\vec{x}^{(t)}$, espaçadas em um intervalo $0 \leq t \leq \tau$. Nesse cenário, o índice i , utilizado anteriormente para identificar as amostras, é substituído pelo passo t , que pode representar um determinado instante no tempo ou uma posição específica em uma série de elementos, dependendo do tipo de problema sendo resolvido.

Para cada passo t , as camadas ocultas recebem como entrada os p atributos das amostras. Assim como na rede neural de propagação direta, as entradas são multiplicadas pela matriz de pesos W_i e alimentam as unidades, produzindo uma seqüência de estados

internos $h^{(t)}$. As saídas dos neurônios das camadas ocultas podem alimentar as entradas de neurônios da mesma camada, de camadas anteriores, de camadas subseqüentes (seguindo o mesmo esquema da propagação direta) ou ainda alimentar as suas próprias entradas. Devido ao alto grau de liberdade nas conexões, diferentes topologias de RNNs podem ser propostas, com maior ou menor complexidade.

Por simplicidade, na Fig. 6, os pesos associados à propagação direta e as funções de ativação foram omitidos do esquema, podendo ser entendidos como componentes do estado $h^{(t)}$ do sistema. Destaca-se, entretanto, apenas o conjunto de pesos das retroalimentações, W_f .

Nesse cenário, a recursividade da rede neural define os estados internos a partir da Eq.3.7:

$$h^{(t)} = g[h^{(t-1)}; \vec{x}^{(t)}; W_i; W_f] \quad (3.7)$$

O comportamento dinâmico das RNNs é facilmente constatado, uma vez que o estado atual h do sistema, em um momento t , é mapeado por uma função g , que tem como argumentos o estado imediatamente anterior, no momento $t - 1$, processado juntamente com uma nova entrada de dados e sob a ação de um conjunto de parâmetros (pesos).

A camada de saída também produz uma seqüência de saídas, $f(t)$. Dependendo do problema, somente a saída final, em $t = \tau$, pode ser relevante para a resolução. Nas RNNs, a camada de saída, por vezes, pode ser entendida como uma camada de leitura (*readout*). O estado atual das camadas ocultas é coletado e convertido em um resultado final por meio do conjunto de pesos de saída, W_o . Nesse sentido, também se mantém o grau de liberdade para selecionar os neurônios das camadas ocultas cujas saídas serão lidas e transformadas em uma predição pelo *readout*.

Pelo esquema da Fig. 6, observa-se que os dados de entrada, que alimentam a rede, permanecem no sistema por um certo tempo, interferindo com os novos dados injetados e produzindo as saídas do sistema. As RNNs apresentam, portanto, uma memória associada, propriedade que as torna a solução ideal para problemas de reconhecimento de voz, predição de séries temporais, tradução ou geração de textos, entre muitas outras possibilidades.

3.2.1 Treinamento de RNNs

Semelhante às redes neurais de propagação direta, o treinamento de RNNs com retropropagação envolve a adaptação dos pesos da rede com base no erro entre a saída predita e o valor real. Entretanto, o processo, neste caso, é realizado por meio do algoritmo Backpropagation Through Time (BPTT), uma extensão da retropropagação de erros tradicional que leva em consideração a natureza sequencial das RNNs [45]. Como as RNNs possuem conexões recorrentes que mantêm memória de estados passados, o BPTT expande

a rede ao longo do histórico de entradas, tratando cada iteração como uma camada da rede. A partir disso, o gradiente do erro é propagado de volta no tempo, ajustando os pesos tanto para os estados atuais quanto para os passados.

Na prática, a recursividade torna os modelos de *deep learning* bastante poderosos. O BPTT deixa claro que, mesmo com a simplicidade das topologias, as redes neurais podem ser representadas de forma eficaz por meio de uma quantidade expressiva de camadas. Essa característica, entretanto, introduz dois problemas comuns durante o treinamento de uma RNN: o desaparecimento e a explosão do gradiente [46].

Na Eq. 3.5, observa-se que o cálculo do gradiente da função de custo depende também do gradiente $a'h(c^{(h)}k)$ da função de ativação dos neurônios da camada. Dependendo da função de ativação escolhida e da entrada da unidade, há uma alta probabilidade de que esse gradiente resulte em um valor muito baixo [47]. Em RNNs, durante a retropropagação em várias camadas, esse fenômeno pode fazer com que o gradiente da função de custo diminua ainda mais, praticamente desaparecendo. Como resultado, independentemente da taxa de aprendizado utilizada, a convergência se torna muito lenta, e o modelo não consegue capturar dependências de eventos distantes na sequência de dados.

Na situação contrária, se ocorrer uma inicialização inadequada dos pesos da rede neural, o gradiente da função de custo pode aumentar indefinidamente, explodindo em valores tão altos que o algoritmo do gradiente enfrentará problemas de convergência, tornando os modelos instáveis.

Atualmente, adaptações dos algoritmos de retropropagação e esquemas mais precisos de inicialização de parâmetros atingiram um estágio avançado de desenvolvimento, sendo altamente eficazes na redução dos efeitos do desaparecimento e explosão do gradiente. No entanto, antes que esses métodos amadurecessem, soluções alternativas de *deep learning* foram propostas para evitar esses problemas, embora sem corrigi-los completamente. Foi nesse contexto que surgiram os modelos de *reservoir computing*, cujo conceito ainda mantém relevância significativa em diversos nichos de pesquisa.

3.3 Reservoir computing

No início do século XXI, as redes de estado de eco (*Echo State Networks*, ESNs) [48] foram introduzidas em publicações científicas, seguidas, pouco tempo depois, pelas máquinas de estado líquido (*Liquid State Machines*, LSMs) [49]. Esses dois tipos de modelos ganharam rapidamente notoriedade por suas demonstrações, que apresentavam, pela primeira vez, as condições matemáticas necessárias para que sistemas dinâmicos aleatórios, em regime estável e, embora próximos do limite da estabilidade, realizassem computações eficazes, operando em tempo discreto.

O conceito geral de *reservoir computing* (RC) surge inspirado por esses estudos.

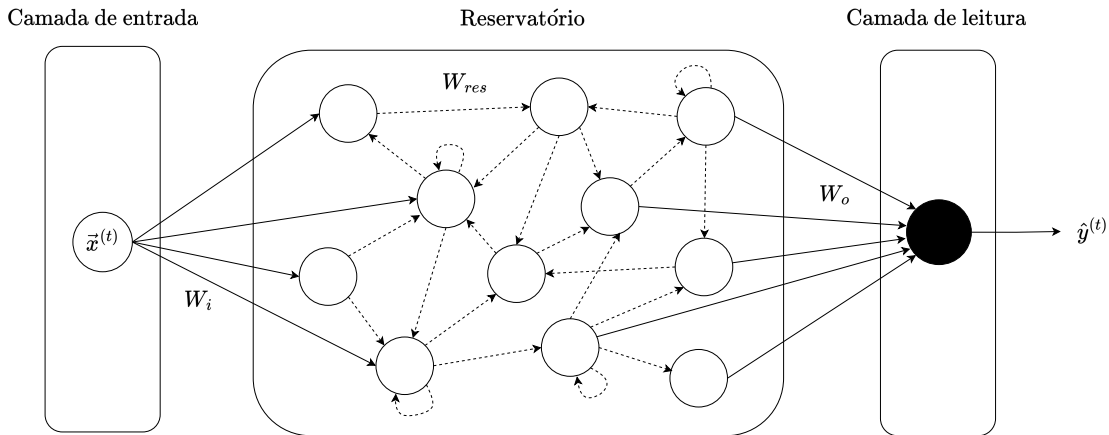


Figura 7 – Topologia ilustrativa de um RC. Os dados sequenciais $\vec{x}^{(t)}$ são alimentados em uma rede recorrente através de uma camada de entrada, cujos pesos de conexão são representados pelo conjunto W_i . Essa camada de entrada é opcional, podendo seus pesos ser simplesmente unitários. Devido à arquitetura dinâmica do RC, as entradas são combinadas com versões anteriores de si mesmas. A totalidade ou uma parte dos neurônios pode ser lida pela camada de saída.

Em um modelo RC, uma RNN é composta por neurônios conectados aleatoriamente, como ilustrado na Fig. 7. A cada passo t , parte do estado $h^{(t)}$ do reservatório é utilizada para gerar uma saída $\hat{y}^{(t)}$, conforme as equações:

$$\begin{aligned} h^{(t)} &= W_i \vec{x}^{(t)} + g_{res} [W_{res} h^{(t-1)}] \\ \hat{y}^{(t)} &= W_o h^{(t)} \end{aligned} \quad (3.8)$$

Onde W_{res} representa a matriz de pesos de conexão das unidades dentro do reservatório, W_o é a matriz de parâmetros associada à camada de leitura e g_{res} é a função de ativação não linear das unidades internas do reservatório.

Como discutido nas seções anteriores, o treinamento de um modelo tradicional consistiria na otimização de todos os parâmetros da rede. No entanto, o paradigma de RC simplifica elegantemente esse processo, reduzindo o treinamento à otimização apenas da matriz de pesos de conexão entre o reservatório e a camada de leitura, W_o . Assim, os parâmetros associados à entrada e ao reservatório são mantidos imutáveis de forma deliberada, o que elimina o problema do desaparecimento ou explosão do gradiente nas RNNs, devido à simplicidade da abordagem. Dessa forma, o treinamento da rede, que originalmente seria uma operação altamente não linear, torna-se um simples problema de regressão linear, conforme indicado na Eq. 3.8.

Como apenas as saídas dos neurônios na camada de leitura são usadas em uma combinação linear para gerar a saída do modelo, o treinamento não afeta a dinâmica interna da rede. Assim, o bloco pode ser caracterizado como um reservatório, cuja função principal é gerar representações diversas das entradas. Essa abordagem é extremamente versátil, pois

permite que um único reservatório seja aplicado a diferentes tipos de problemas, bastando treinar camadas de leitura distintas para extrair diferentes atributos do reservatório.

Na prática, a rede recorrente atua como um bloco de pré-processamento, que transforma entradas de pequena dimensão em estados de um espaço de alta dimensão de forma não linear, devido ao grande número de neurônios coletados na camada de leitura e à dinâmica interna da rede.

A propriedade de mapeamento das entradas em um espaço de dimensão mais alta confere ao RC a capacidade de separar dados linearmente de forma mais trivial [50]. Retornando ao problema da operação XOR, discutido na Seção 2.2.2.2, verificou-se que o mapeamento bidimensional dos resultados (Fig. 4b) não era linearmente separável, o que tornava a classificação bastante desafiadora. Na Fig. 8, a introdução inteligente de uma terceira dimensão nos dados de entrada possibilita a definição de um plano linear bidimensional, que atua como uma fronteira entre os dois espaços de decisão. Assim, essa capacidade de mapeamento é compreendida como fundamental para a eficácia do RC na resolução de problemas de classificação mais complexos.

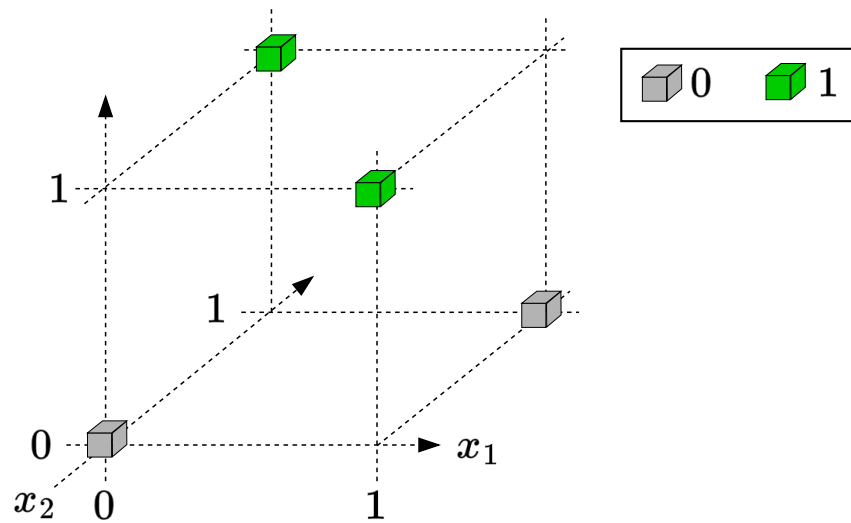


Figura 8 – Operação lógica XOR mapeada tridimensionalmente. As variáveis apresentadas na Fig. 4b mantêm suas posições iniciais nos eixos das abscissas e ordenadas. No entanto, os cubos verdes (1) são reposicionados ao longo do eixo das cotas em comparação com os cubos cinzas (0).

Em termos de comportamento dinâmico, o reservatório deve possuir uma capacidade de memória limitada. Isso significa que o estado atual do reservatório deve ser influenciado por entradas recentes, mas deve ser independente de entradas de um passado distante. É essencial que o reservatório apresente a propriedade de memória desvanecente (*fading memory*), ou seja, o estado da rede se torne assintoticamente dissociado das condições iniciais. Além disso, o reservatório precisa ser suficientemente dinâmico para produzir uma ampla variedade de representações dos dados de entrada. Dessa forma, pequenas

variações nas entradas devem levar a trajetórias distintas de estados dentro do reservatório, possibilitando que o sistema diferencie diversos padrões.

Embora pouco mais de duas décadas de pesquisas em RC tenham sido realizadas, o conhecimento sobre como alinhar com precisão o comportamento dinâmico dos reservatórios às dinâmicas dos problemas ainda seja bastante limitado [51]. Em geral, a escolha de um reservatório exige a identificação de um sistema cujas dinâmicas intrínsecas correspondam às exigências específicas do problema. Felizmente, o fato de o reservatório não precisar ser treinado torna diversos sistemas candidatos ideais para esse paradigma, incluindo sistemas físicos, como circuitos fotônicos.

4 RESERVOIR COMPUTING FOTÔNICO

O *reservoir computing* fotônico (PRC) representa uma abordagem inovadora no campo da computação neuromórfica, combinando as vantagens das redes neurais recorrentes com a velocidade e eficiência da tecnologia fotônica. Ao utilizar a luz como meio de propagação de informações, o PRC oferece promessas significativas para aplicações em tempo real, processamento paralelo e computação de alta velocidade, posicionando-se como uma alternativa promissora para a superação de limitações energéticas e de escalabilidade encontradas em sistemas puramente eletrônicos.

Uma das vantagens do PRC é a possibilidade de remover a não linearidade do interior do reservatório, transferindo-a para uma operação não linear na camada de leitura, utilizando metodologias de medição próprias dos sistemas ópticos. O estudo teórico de [52] aborda como a implementação de uma camada de saída não linear pode superar algumas limitações impostas pela simples combinação linear da Eq. 3.8, permitindo uma exploração mais eficaz da complexidade dimensional dos sinais gerados pelo reservatório.

$$\hat{y}^{(t)} = g_o[W_o h^{(t)}] \quad (4.1)$$

Em sistemas fotônicos que operam com luz coerente, a não linearidade de g_o pode ser atendida de forma simples pelo comportamento quadrático g_{det} dos fotodetectores. Esses dispositivos convertem a amplitude dos campos de saída dos neurônios do reservatório em níveis quantizados de potência real através da leitura de fótons emitidos. O sinal resultante é, então, utilizado como entrada da camada de leitura para realizar a regressão linear, conforme descrito na Eq. 4.2.

$$\begin{aligned} h^{(t)} &= W_i \vec{x}^{(t)} + W_{res} h^{(t-1)} \\ \hat{y}^{(t)} &= W_o g_{det}[h^{(t)}] \end{aligned} \quad (4.2)$$

Os sensores de medição, embora facilitem a implementação dos reservatórios, frequentemente impõem limitações à banda de operação do PRC devido à velocidade dos aparelhos de medição e dos geradores de sinal. No entanto, essa característica dos sistemas ópticos permite a implementação de computadores neuromórficos passivos, nos quais não é necessário consumir potência dentro dos reservatórios para manter a não linearidade.

É evidente que a flexibilidade e as características dos sistemas fotônicos abrem caminho para diversas abordagens na computação neuromórfica. Essa adaptabilidade leva à emergência de duas amplas classes de PRC, cada uma com suas especificidades e potencialidades.

4.1 Reservatórios fotônicos integrados em paralelo

Aproveitando o potencial da fotônica em termos de largura de banda e consumo de energia, as propostas iniciais de reservatórios dessa classe utilizavam componentes ópticos integrados em chips. Esses componentes eram conectados de maneira a emular, o mais próximo possível, as ESNs e suas não linearidades, processando as entradas de forma distribuída e paralela.

Nesse contexto, as primeiras topologias de reservatórios fotônicos integrados eram formadas por malhas de amplificadores ópticos semicondutores (SOAs) conectados [53]. O conceito central consistia em replicar as funções de ativação em tangente hiperbólica, comumente utilizadas em alguns algoritmos de computação por reservatório, a partir do comportamento não linear dos dispositivos ópticos integrados no sistema.

Posteriormente, em [54], foi apresentada uma malha composta por 16 núcleos, constituídos por guias de onda em espiral, divisores e combinadores ópticos (MMI). O funcionamento desse sistema baseou-se exclusivamente na interferência entre o sinal de entrada e suas versões passadas, que coexistiam temporalmente dentro do reservatório. Essa abordagem demonstrou com sucesso a transferência da não linearidade do reservatório para os fotossensores, resultando em uma arquitetura desenvolvida apenas com componentes passivos. Essa característica garante um compromisso com a eficiência energética durante o processamento.

Devido às limitações de conexão entre as unidades das topologias anteriores, foi proposta, em [55], uma cavidade em cristal fotônico como reservatório. Nessa arquitetura, a luz é enviada e recolhida por meio de guias de onda. Dentro da cavidade, os campos eletromagnéticos interagem continuamente entre si, resultando em um comportamento caótico e complexo nas interconexões. Esse comportamento foi explorado, em simulações, para realizar operações lógicas como XOR e AND, além do reconhecimento do cabeçalho de um pacote de bits.

Paralelamente, outras arquiteturas foram propostas no campo dos reservatórios fotônicos integrados, fazendo uso de propriedades de ressonância em cavidades cristalinas [56] e em redes de anéis ressoadores [57].

4.2 Reservatórios multiplexados no tempo

Conforme consolidado por [58], o conceito de reservatórios multiplexados por divisão de tempo simplifica as redes compostas por múltiplas unidades interconectadas, reduzindo-as a um único neurônio altamente não linear com retroalimentação em atraso. Nesse contexto, a topologia de reservatório apresentada na Fig. 7, na qual todas as unidades são teoricamente acessíveis externamente, passa a ser equivalente à arquitetura ilustrada na Fig. 9.

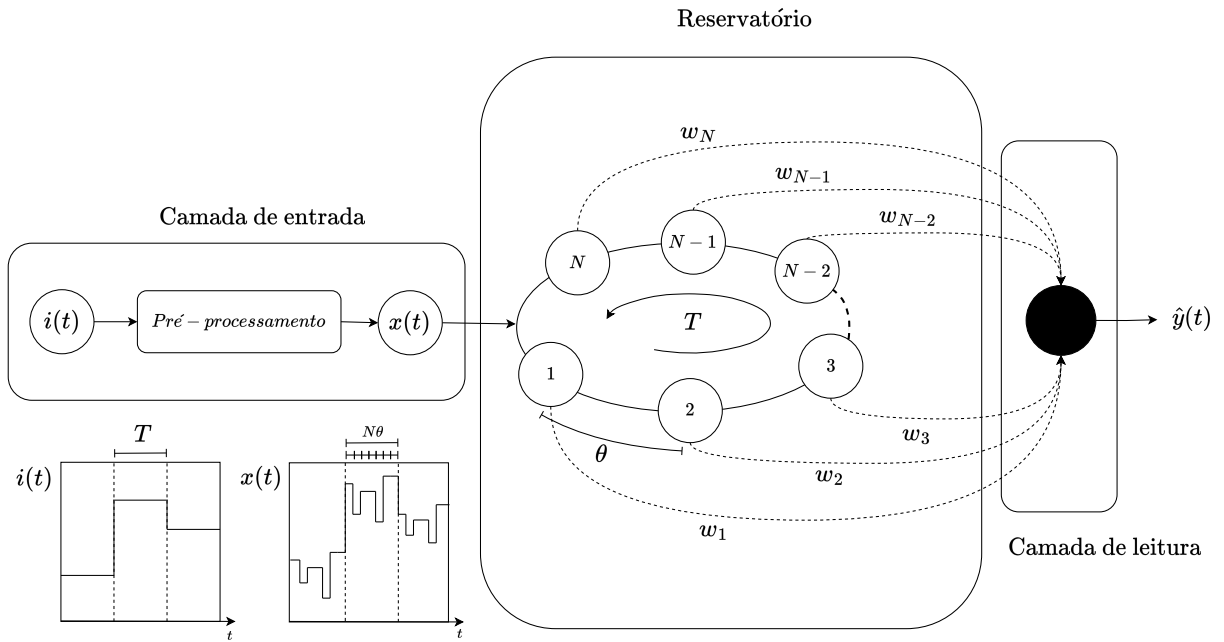


Figura 9 – Esquema de um reservatório multiplexado no tempo, onde todas as entradas alimentam um único neurônio não linear. Para simplificação, a entrada é representada como uma variável temporal unidimensional na figura; no entanto, este esquema é aplicável a entradas de qualquer dimensão p .

Para compensar a perda de paralelismo em relação aos reservatórios integrados, a entrada precisa passar por um pré-processamento. Inicialmente, a sequência de entrada, seja em tempo contínuo ou discreto, deve passar por um circuito S/H , resultando na forma de onda $i(t)$, que mantém o valor da entrada por um período T . Em seguida, para introduzir maior complexidade nos estados gerados pelo sistema, uma máscara $M(t)$ é aplicada a $i(t)$, produzindo, conforme a Eq. 4.3, a entrada $x(t)$ que será injetada na unidade não linear.

$$x(t) = M(t) \cdot i(t) \quad (4.3)$$

A máscara $M(t) = M(t + T)$ é uma função T -periódica, definida em partes, onde cada um dos seus N segmentos assume valores constantes ao longo dos intervalos $(i - 1)\theta \leq t \leq i\theta, \forall, i \in \mathbb{N}^*$. Os valores atribuídos a cada segmento, em seu respectivo intervalo de duração θ , podem ser escolhidos de forma aleatória.

Dentro de um esquema de serialização, a máscara define N pontos equidistantes dentro do intervalo T , com separação temporal de $\theta = T/N$. Cada um desses pontos pode ser interpretado como um neurônio virtual da rede, de maneira que o estado do reservatório é caracterizado pelos valores de saída em cada um desses neurônios virtuais. Ademais, os valores dos segmentos da máscara podem ser considerados como os pesos de entrada associados a cada neurônio virtual [59].

Em relação à dinâmica de funcionamento, as entradas das unidades virtuais são sequencialmente injetadas no neurônio não linear e precisam ser armazenadas na rede. Nesse esquema, o reservatório deve garantir uma retroalimentação com atraso suficiente para que o estado de uma unidade virtual influencie, ao longo do tempo, o estado das outras unidades e dela própria. Para enriquecer a variabilidade de estados no reservatório, o sistema deve operar em regime transiente. Assim, considerando τ como a escala de tempo do sistema, se $\tau \ll \theta$, as entradas de cada unidade virtual são mantidas constantes o suficiente para que o sistema atinja o regime estacionário. Nessa situação, os neurônios virtuais não se acoplam entre si. No entanto, quando $\tau > \theta$, o sistema permanece em regime transiente, o que resulta no autoacoplamento dos neurônios virtuais e no acoplamento entre eles e seus vizinhos. Os pesos dessas conexões tendem a diminuir exponencialmente ao retroceder no tempo, em conformidade com a propriedade de memória desvanescente do RC. De modo geral, para garantir um bom desempenho do PRC, busca-se manter $\theta \lesssim \tau \ll T$. Além disso, o número N de unidades virtuais não deve ser tão grande a ponto de comprometer a resposta do sistema às entradas e à retroalimentação, preservando a capacidade do reservatório de processar as informações de maneira eficiente.

Para a leitura dos estados do reservatório e a solução de um problema específico, o algoritmo de aprendizagem extrai a saída f dos neurônios virtuais, conforme o esquema de multiplexação. No domínio da fotônica, essa extração é realizada com medidas de intensidade (fotodetecção) dos campos eletromagnéticos de saída dos circuitos físicos de reservatório. Em seguida, um peso de saída é atribuído a cada neurônio virtual em *software*, de modo que a combinação linear dos estados obtidos aproxima a saída desejada por:

$$\hat{y}(t) = \sum_{i=1}^N w_i \cdot f[T(t-1) + i\theta] \quad (4.4)$$

Baseando-se nesse esquema de RC, o trabalho de [60] introduz o primeiro reservatório optoeletrônico, capaz de realizar tarefas como reconhecimento de fala e equalização de canal. Os resultados desse estudo também evidenciam a flexibilidade da arquitetura de reservatório multiplexado no tempo, indicando que o conceito pode ser aplicado em sistemas que não possuem uma escala de tempo intrínseca.

Recentemente, demonstrou-se que um microrressonador em anel pode operar como um sistema de retroalimentação com atraso, sendo útil para um reservatório multiplexado no tempo [28]. A mesma equipe de pesquisa envolvida no estudo tem investigado adaptações e características não lineares do dispositivo. Os resultados obtidos confirmam que o microrressonador em anel é uma plataforma extremamente compacta e eficiente em comparação com as outras arquiteturas de PRC apresentadas nesta seção.

5 MICRORRESSONADORES EM ANEL

Microrressonadores em anel (MRR) são estruturas ópticas de pequeno porte que desempenham um papel fundamental em diversas aplicações, como telecomunicações, sensores e sistemas de processamento de sinais. Esses dispositivos utilizam o princípio da ressonância para armazenar e acumular a energia luminosa em comprimentos de onda específicos, permitindo um controle preciso das propriedades ópticas. Além disso, a possibilidade de miniaturização desses componentes em escala micrométrica pela plataforma de silício contribuiu para sua integração em circuitos fotônicos, abrindo caminho para avanços e inovações em tecnologias de fotonica integrada.

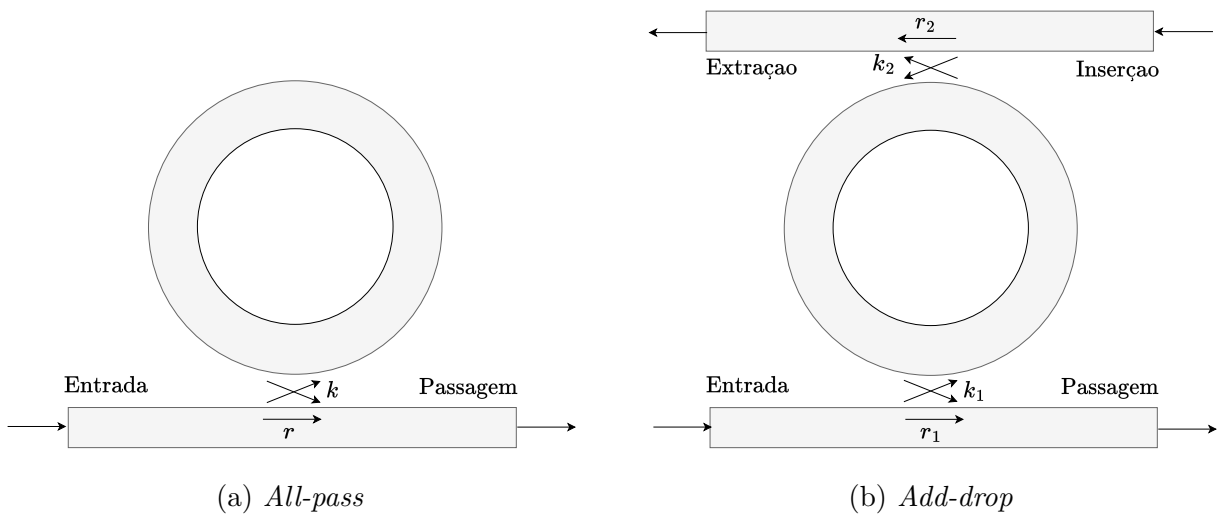


Figura 10 – Configurações clássicas de MRR

A estrutura de um MRR é formada ao conectar as extremidades de um guia de onda, criando um circuito fechado que pode ser acessado através de um sistema de acoplamento. Isso permite que o dispositivo apresente ressonâncias quando o caminho óptico tiver uma extensão L equivalente a um múltiplo inteiro do comprimento de onda λ da luz incidente no anel. Em outras palavras, quando a onda sofre um deslocamento de fase φ correspondente a um múltiplo inteiro m de 2π radianos, ocorre interferência construtiva dentro do anel. Assim:

$$\varphi = \beta L = 2\pi m \implies \lambda_0 = \frac{n_{eff} L}{m}, \quad \forall m \in \mathbb{N}^* \quad (5.1)$$

onde λ_0 é o comprimento de onda ressonante, $\beta = 2\pi n_{eff}/\lambda$ é a constante de propagação e n_{eff} é o índice de refração efetivo do modo propagante no anel.

Os MRRs comumente apresentam um caminho óptico circular, mas também existem versões com geometria alongada. Essas variações são compostas por seções retas conectadas

por curvas semicirculares nas extremidades, formando uma estrutura semelhante a uma pista de corrida, o que lhes dá o nome de ressonadores *racetrack*. Devido a esse design, o ressonador *racetrack* oferece maior flexibilidade para ajustar o comprimento efetivo do caminho óptico, facilitando o controle de parâmetros como ressonância, fator de qualidade e acoplamento [61].

Como apresentado na Fig. 10, tradicionalmente, dois arranjos de MRR são possíveis [62].

5.1 All-pass

A configuração mais básica de um MRR, conhecida como *all-pass* (cf. Fig. 10a), é formada conectando uma das saídas de um acoplador direcional a uma de suas entradas por meio de um guia de onda.

Quando a condição de ressonância é satisfeita, as ondas injetadas na porta de entrada do barramento são acopladas ao anel por meio do campo evanescente. Nesse contexto, os coeficientes de transmissão r e de acoplamento cruzado k descrevem, respectivamente, a quantidade de luz que se acopla dentro do mesmo canal e a quantidade que é transferida entre canais diferentes no acoplador direcional. Assumindo que não há perdas na região de acoplamento, vale a relação $r^2 + k^2 = 1$.

Após a transferência da luz do barramento para o anel, a dinâmica da estrutura permite que a luz seja novamente acoplada ao barramento e transmitida à porta de passagem ou, alternativamente, que permaneça dentro do anel. Dessa forma, a energia pode ser temporariamente armazenada dentro do ressonador. A Eq. 5.2 define a relação fasorial entre o campo E_p transmitido na saída do ressonador *all-pass* e o campo E_i de entrada.

$$\frac{E_p}{E_i} = \frac{r - ae^{j\varphi}}{1 - rae^{j\varphi}} = e^{j(\pi+\varphi)} \frac{a - re^{-j\varphi}}{1 - rae^{j\varphi}} \quad (5.2)$$

O termo $a = e^{-\frac{\alpha L}{2}}$, onde α é o coeficiente de atenuação do anel, representa a fração da amplitude que é transmitida diretamente através do sistema acoplador após interagir uma única vez com o ressonador, antes que o sinal possa recircular ou interferir devido ao acoplamento contínuo. Detona-se por αL a fração da potência perdida a cada ciclo no anel por propagação e acoplamento.

Dessa maneira, a função de transferência T_p^{all} , que descreve a razão da potência transmitida para a porta de passagem pela potência incidente na porta de entrada, depende do coeficiente de atenuação do anel α e do coeficiente de transmissão, conforme a Eq. 5.3.

$$T_p^{all} = \left| \frac{E_p}{E_i} \right|^2 = \frac{a^2 - 2ra \cos \varphi + r^2}{1 - 2ra \cos \varphi + (ra)^2} \quad (5.3)$$

Para uma cavidade com atenuação pouco significativa, em que $a \approx 1$, observa-se que a transmissão será unitária para qualquer valor de frequência ou comprimento de onda. Nessa condição, a estrutura atua como um filtro passa-tudo, o que justifica sua denominação.

A defasagem efetiva ϕ adquirida pela luz ao percorrer o ressonador, partindo da porta de entrada até a porta de passagem, pode ser determinada por:

$$\phi = \arg\left(\frac{E_p}{E_e}\right) = \pi + \varphi^* + \text{tg}^{-1}\left(\frac{r \text{sen}\varphi^*}{a - r \text{cos}\varphi^*}\right) + \text{tg}^{-1}\left(\frac{r a \text{sen}\varphi^*}{1 - r a \text{cos}\varphi^*}\right) \quad (5.4)$$

onde

$$\varphi^* = \frac{2\pi n_g L(\lambda - \lambda_0)}{\lambda_0^2} \quad (5.5)$$

é a defasagem normalizada em $[-2\pi, 2\pi]$ com relação à ressonância. O termo n_g é o índice de grupo.

Dependendo da relação entre a atenuação e o coeficiente de transmissão, é possível estabelecer três regimes de funcionamento para o sistema. Nesse sentido, quando a potência acoplada é equivalente à potência atenuada no MRR, ou seja, $r = a$, tem-se uma situação de acoplamento crítico, onde a transmissão no comprimento de onda ressonante é nula. Para $r < a$, diz-se que ocorre superacoplamento. Para $r > a$, diz-se que ocorre subacoplamento. As transmissões características de cada um dos regimes de um ressonador foram demonstradas experimentalmente em [63].

5.2 Add-drop

Ao adicionar um segundo barramento à estrutura do ressonador *all-pass*, cria-se um novo caminho de propagação para a luz, que agora pode ser parcialmente transmitida para uma nova porta de extração (Fig. 10b). Essa configuração é conhecida como *add-drop*, e suas funções de transmissão nas portas de passagem e extração são dadas por:

$$T_p^{add} = \left|\frac{E_p}{E_e}\right|^2 = \frac{r_2^2 a^2 - 2r_1 r_2 a \cos \varphi + r_1^2}{1 - 2r_1 r_2 a \cos \varphi + (r_1 r_2 a)^2} \quad (5.6)$$

$$T_{ex}^{add} = \left|\frac{E_{ex}}{E_e}\right|^2 = \frac{a(1 - r_1^2)(1 - r_2^2)}{1 - 2r_1 r_2 a \cos \varphi + (r_1 r_2 a)^2} \quad (5.7)$$

Para acopladores balanceados ou simétricos, tem-se que $r_1 = r_2$. Nessa situação, se a atenuação do anel for desprezível, $a \approx 1$, ocorre o acoplamento crítico.

5.3 Parâmetros de desempenho

As configurações de MRR podem ser descritas por características espectrais específicas, que dependem dos coeficientes de acoplamento e das perdas por atenuação [18]. Como os MRRs operam como filtros frequenciais, esses parâmetros são geralmente figuras de mérito amplamente utilizadas no campo dos filtros ópticos.

A partir das funções de transferência de transmissão, é possível determinar a largura total na metade do máximo (FWHM) do espectro de ressonâncias, que é dada por:

$$\text{FWHM} = \frac{(1 - ra)\lambda_0^2}{\pi n_g L \sqrt{ra}} \quad (5.8)$$

Para a configuração *add-drop*, tem-se $r = r_1 r_2$.

A faixa espectral livre (FSR), que descreve a distância em comprimentos de onda entre duas ressonâncias do espectro, é dada por:

$$\text{FSR} = \frac{\lambda_0^2}{n_g L} \quad (5.9)$$

Tanto no FWHM, quanto no FSWR utiliza-se o índice de grupo n_g para levar em consideração a dispersão em primeira ordem do guia de onda. Esse parâmetro é definido por:

$$n_g = n_{eff} - \lambda_0 \frac{dn_{eff}}{d\lambda} \quad (5.10)$$

Utilizando as características espectrais definidas acima, define-se duas outras medidas qualitativas dos MRRs: a finesse \mathcal{F} (Eq. 5.11) e o fator de qualidade Q (Eq. 5.12).

$$\mathcal{F} = \frac{\text{FSR}}{\text{FWHM}} \quad (5.11)$$

$$Q = \frac{\lambda_0}{\text{FWHM}} \quad (5.12)$$

Como o espectro de transmissão das configurações apresenta vales de ressonância, a finesse \mathcal{F} é uma medida da precisão da ressonância com relação ao espaçamento entre os vales, enquanto que o fator de qualidade Q é uma medida da precisão da ressonância com relação à sua frequência central.

O significado físico do fator Q está intimamente ligado à relação entre a energia circulante armazenada no ressonador e a energia dissipada a cada ciclo no anel. Seja T_R o tempo dispendido para a luz completar um ciclo no ressonador, definindo N_{ciclos} como o

número de ciclos necessários para reduzir a energia circulante para $1/e$ do seu valor inicial, encontra-se que o fator de qualidade é dado por:

$$Q = \left(\frac{2\pi c}{\lambda_0} \right) \cdot T_R \cdot N_{ciclos} \quad (5.13)$$

onde c representa a velocidade da luz no vácuo. De forma análoga, a finesse \mathcal{F} representa um fator de 2π do número de ciclos N_{ciclos} realizados pela luz no ressonador até o decaimento da energia no patamar estabelecido.

Devido ao alto confinamento nos guias de ondas dos microrressonadores em anel, essas estruturas apresentam alguns efeitos não lineares. Uma discussão mais aprofundada a respeito foge, entretanto, ao escopo do projeto.

6 DESENVOLVIMENTO METODOLÓGICO

O comportamento de dispositivos ópticos apresenta elevada sensibilidade a variações de temperatura, o que pode impactar diretamente a eficiência e a precisão no processamento de informações em sistemas fotônicos.

Neste capítulo, serão apresentados os métodos e procedimentos adotados para demonstrar e analisar como as flutuações térmicas afetam o desempenho de um sistema RC implementado com um MRR na configuração *all-pass*. Para investigar esse fenômeno, explora-se a relação entre as propriedades térmicas e ópticas do ressonador, destacando seu efeito sobre a ressonância e a estabilidade do sistema.

A metodologia desenvolvida baseia-se principalmente na modelagem óptica por meio de uma ferramenta *open-source* de design, otimização e simulação de circuitos fotônicos.

Esta abordagem compreende, como ponto de partida, a descrição e a caracterização das propriedades do ressonador em anel, implementado em *software*. O comportamento do MRR projetado é analisado no domínio do tempo e da frequência para melhor compreensão do sistema neuromórfico.

Em seguida, realiza-se uma prova de conceito, na qual o circuito fotônico proposto é utilizado como um reservatório apto a resolver um problema não trivial de *machine learning*: a operação XOR. O desempenho do RC será, então, examinado em um ambiente com variações de temperatura. Uma análise adicional de parâmetros de acoplamento visa compreender como esses fatores impactam a precisão no processamento de dados pelo sistema. Com isso, busca-se estabelecer um modelo robusto que permita prever e ajustar o comportamento do ressonador frente às flutuações térmicas, garantindo, assim, maior estabilidade e eficiência ao sistema.

6.1 Photontorch

As simulações necessárias foram realizadas utilizando o Photontorch, um framework desenvolvido em Python que oferece um conjunto de ferramentas para a simulação e otimização de circuitos fotônicos nos domínios do tempo e da frequência [64].

6.1.1 Formalismo tensorial

Geralmente, para sistemas óticos mais simples e menores, como guias de onda modelados por estruturas em camadas, as soluções das equações de onda obtidas a partir das equações de Maxwell são suficientes para determinar analiticamente os modos de propagação e os respectivos índices de refração efetivos. Para sistemas mais complexos, outros métodos numéricos são frequentemente necessários para as análises, como é o caso

do método de diferenças finitas no domínio do tempo (FDTD), que discretiza as equações de Maxwell em uma malha.

Em circuitos fotônicos, onde diversos componentes ópticos interagem entre si, a complexidade do sistema pode, em certos casos, resultar em simulações numéricas muito custosas. Nesse contexto, o pacote Photontorch utiliza o formalismo de matrizes de espalhamento (S) como uma abordagem para simplificar a modelagem de um circuito completo, permitindo prever seu comportamento a partir das propriedades de seus componentes isolados [65].

Basicamente, cada componente n é representado por uma “caixa preta” com P portas de entrada e Q portas de saída. A relação entre um estado de entrada \vec{x}_i e o de saída \vec{x}_o em um instante $t + dt$ é determinada por uma matriz $S_n \in \mathbb{C}^{2Q \times 2P}$, conforme a Eq. 6.1.

$$\vec{x}_o^{(n)}(t + dt) = S_n \cdot \vec{x}_i^{(n)}(t) \quad (6.1)$$

Em outras palavras, cada elemento S_{ij} da matriz de espalhamento relaciona o campo de entrada na porta i com o campo resultante na porta j , seguindo o comportamento do componente. Supondo uma propagação bidirecional dos campos eletromagnéticos nos componentes, a matriz S é considerada simétrica.

Nesse âmbito, para projetar um circuito completo em Photontorch, cada um dos N_c componentes lineares de um determinado sistema pode se interconectar consigo ou com outros, como blocos de *design*, de forma que suas matrizes individuais S_n , com $n \in [1, N_c]$, são combinadas em uma matriz conjunta S , como apresentado na Eq. 6.2.

$$S = \begin{bmatrix} S_1 & & & \\ & S_2 & & \\ & & \ddots & \\ & & & S_{N_c} \end{bmatrix} \quad (6.2)$$

Dessa forma, seja $\{\vec{x}_i^{(n)}\}_{n=1, \dots, N_c}$ o conjunto de vetores associados aos campos de entrada. O conjunto de saída resultante do sistema coletivo é determinado por:

$$\vec{x}_o = \begin{bmatrix} \vec{x}_o^{(1)} \\ \vec{x}_o^{(2)} \\ \vdots \\ \vec{x}_o^{(N_c)} \end{bmatrix} = \begin{bmatrix} S_1 & & & \\ & S_2 & & \\ & & \ddots & \\ & & & S_{N_c} \end{bmatrix} \begin{bmatrix} \vec{x}_i^{(1)} \\ \vec{x}_i^{(2)} \\ \vdots \\ \vec{x}_i^{(N_c)} \end{bmatrix} \quad (6.3)$$

A partir da similaridade entre as Eqs. 6.1 e 6.3, observa-se que a biblioteca Photontorch interpreta circuitos completos como novos blocos de *design*, permitindo a

hierarquização de diferentes níveis de circuitos. As conexões internas entre as portas dos componentes de um circuito são definidas pela matriz de interconexão C_{int} , enquanto as portas dos componentes que não possuem conexões internas tornam-se portas externas, possibilitando a conexão do circuito com outros circuitos externos por meio da matriz de conexão C_{ext} . Os elementos de ambas as matrizes pertencem ao conjunto $0, 1$.

Assim, a entrada geral (*top-level*) do circuito \vec{x}_i^{ext} e a saída geral do circuito \vec{x}_o^{ext} se relacionam pela Eq. 6.4 a seguir, de forma independente dos estados internos do bloco.

$$\vec{x}_o^{ext} = (C_{ext})^T \cdot S \cdot (I - C_{int} \cdot S)^{-1} \cdot C_{ext} \cdot \vec{x}_i^{ext} \quad (6.4)$$

I é a matriz identidade.

A Eq. 6.4 representa o método principal para resolver um circuito fotônico no domínio da frequência no Photontorch. No entanto, essa forma simplificada não é precisa no domínio do tempo, pois não considera os atrasos temporais presentes no circuito. Para sistemas com dependência temporal, o Photontorch inicializa a instância do circuito reorganizando as portas, classificando-as em portas sem memória (ML) e portas com memória (MC). As portas ML dependem exclusivamente das operações com matrizes de espalhamento, enquanto as portas MC correspondem às demais portas dos componentes que apresentam atraso temporal ou decorrem de fontes ou detectores [66]. Novas matrizes de conexão são produzidas, relacionando a conexão e interconexão entre portas MC e portas ML. Além disso, geram-se novas matrizes de espalhamento para cada tipo de porta. Assim, o comportamento temporal de um circuito pode ser determinado pela solução da Eq. 6.5.

$$\vec{x}_o^{ext}(t + dt) = \left[C^{MCMC} + C^{MCML} \cdot S^{MLML} \cdot (1 - C^{MCML} S^{MLML})^{-1} \cdot C^{MCML} \right] \cdot S^{MCMC} \cdot \vec{x}_i^{ext}(t) \quad (6.5)$$

As Eqs. 6.4 e 6.5 indicam que as soluções em simulação envolvem principalmente operações com tensores. Para otimizar a manipulação dessas equações, o Photontorch é implementado a partir da biblioteca PyTorch, que permite a execução de simulações paralelas, inclusive em GPU, e a otimização de parâmetros do sistema por retropropagação de erros.

Neste projeto, utilizou-se a versão Photontorch 0.4.1 [67], que possui uma biblioteca nativa composta pelos principais componentes óticos, como guias de onda, interferômetros multimodo, acopladores, entre outros. Os códigos foram implementados e executados com a distribuição Python 3.11.0. A precisão numérica foi determinada por uma representação em ponto flutuante de precisão simples (32 bits).

A seguir, descreve-se como o circuito foi projetado utilizando as ferramentas disponíveis no pacote Photontorch.

6.1.2 Esquemático e equacionamento do circuito

Conforme discutido no Capítulo 5, a configuração *all-pass* de um microrressonador em anel (*mrr*) pode ser representada por um acoplador direcional (*dc*) conectado a si mesmo por meio de um guia de onda (*wg*), conforme ilustrado na Fig. 11.

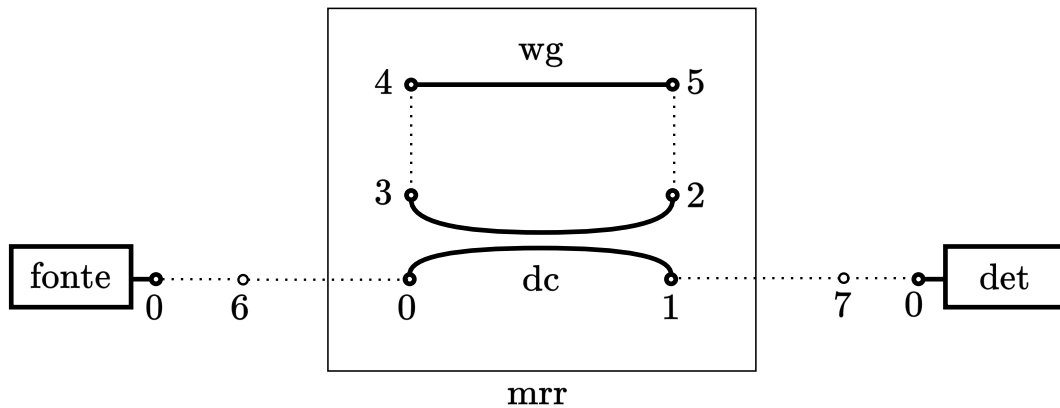


Figura 11 – Esquemático do circuito implementado.

No bloco *mrr*, as portas internas 2 e 3 do acoplador direcional conectam-se, respectivamente, às portas 5 e 4 do guia de onda. As portas internas 0 e 1 do acoplador permanecem livres para conexão com componentes externos ao bloco, tornando-se equivalentes às portas externas 6 (entrada) e 7 (passagem). Dessa forma, o circuito é completado com a adição de uma fonte e de um fotodetector, conectados a essas portas.

Utilizando o formalismo da seção anterior, a matriz S do guia de onda é representada pela Eq. 6.6, considerando a propagação no sentido positivo do eixo.

$$S_{wg} = 10^{-\alpha_{wg} L/20} \begin{bmatrix} 0 & e^{-(j\beta L + \theta)} \\ e^{-(j\beta L + \theta)} & 0 \end{bmatrix} \quad (6.6)$$

onde α_{wg} é o fator de perda por propagação no guia de onda em dB/m, L é o comprimento do guia e θ é um fator adicional de correção de fase, assumido como nulo. O comportamento do guia de onda corresponde essencialmente a uma defasagem no campo incidente em uma das portas do componente, juntamente com uma possível atenuação, ambas resultando em uma alteração na amplitude da luz transmitida.

Já o comportamento de acoplamento evanescente do acoplador direcional é representado pela matriz de espalhamento da Eq. 6.7, conforme a disposição ordenada das portas no esquemático da Fig.11.

$$S_{dc} = \begin{bmatrix} 0 & r & jk & 0 \\ r & 0 & 0 & jk \\ jk & 0 & 0 & r \\ 0 & jk & r & 0 \end{bmatrix} \quad (6.7)$$

Assume-se que o acoplador direcional não tenha comprimento, atenuação ou lacuna entre os dois guias de onda que o compõem.

Nessas condições, a matriz S do bloco mrr é obtida por meio da Eq. 6.2, combinando S_{wg} e S_{dc} , o que resulta em:

$$S = \begin{bmatrix} 0 & r & jk & 0 & 0 & 0 \\ r & 0 & 0 & jk & 0 & 0 \\ jk & 0 & 0 & r & 0 & 0 \\ 0 & jk & r & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \Phi \\ 0 & 0 & 0 & 0 & \Phi & 0 \end{bmatrix} \quad (6.8)$$

onde $\Phi = 10^{-\alpha_{wg} L/20} e^{-j\beta L}$. Considerando a disposição das portas do esquemático, as matrizes de conexão são dadas por:

$$C_{int} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \quad (6.9)$$

$$C_{ext} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \quad (6.10)$$

A matriz C_{ext} pode ser entendida como uma extensão em duas colunas à direita da matriz C_{int} , que relaciona como as 6 portas internas do bloco estariam conectadas às duas portas externas.

A partir da Eq. 6.4, o comportamento do ressonador em anel no domínio da frequência pode ser resumido na matriz de espalhamento geral:

$$S_{mrr} = \begin{bmatrix} 0 & \frac{r-\Phi}{1-r\Phi} \\ \frac{r-\Phi}{1-r\Phi} & 0 \end{bmatrix} \quad (6.11)$$

Observando a forma de S_{mrr} , identifica-se que a relação entre a amplitude dos campos na porta de passagem e na porta de entrada é dada por:

$$\frac{E_p}{E_i} = \frac{E_7}{E_6} = \frac{r - \Phi}{1 - r\Phi} \quad (6.12)$$

Assim, assumindo que $\alpha_{wg} = \frac{10 \alpha}{\ln 10}$, a resposta de transmissão em frequência seria dada por:

$$T_p^{all}(s) = \left(\frac{r - ae^{-j\beta L}}{1 - rae^{-j\beta L}} \right) s \quad (6.13)$$

Verifica-se que o módulo ao quadrado da Eq. 6.13 conduz ao mesmo resultado apresentado na Eq. 5.3 para a transmissão do microrressonador *all-pass*.

6.2 Caracterização do ressonador em anel e análise comportamental

6.2.1 Parâmetros de design

O projeto do circuito é concluído com a definição dos parâmetros de cada componente do bloco *mrr* (cf. Tab. 2).

Parâmetro	Valor
L	$2\pi \times 53.00 \mu\text{m}$
α_{wg}	4,00 dB/cm
n_g	3,61
n_{eff}	2,79
λ_0	1550,00 nm
k	0,31
r	0,95

Tabela 2 – Valores dos parâmetros utilizados na simulação do circuito da Fig. 11.

O circuito foi projetado para atuar na banda C com ressonância em 1550 nm, comprimento de onda para o qual o silício apresenta experimentalmente o índice de grupo n_g da Tab. 2 [68]. Considera-se que a geometria do guia de onda e a estrutura do componente conduzem a um índice de refração efetivo com confinamento intermediário dentro da banda de operação do ressonador.

Por simplicidade, foi escolhido um MRR com anel circular de raio $53 \mu\text{m}$, de forma que o T_R do ressonador fosse próximo de 4 ps. As perdas do sistema foram limitadas à atenuação no guia de onda, desconsiderando-se as perdas por curvatura. Nesse contexto, como valor de referência para a perda por propagação, foram utilizados resultados experimentais com microrressonadores em anel compactos, operando na banda de frequências selecionada, que apresentaram baixíssimas perdas de propagação na plataforma SOI [69].

Desprezando igualmente a retrodifusão, o comportamento do ressonador é determinado pelo coeficiente $a = 0,98$. O coeficiente de acoplamento cruzado $k^2 = 0,10$

foi estabelecido para que o ressonador estivesse no modo de operação convencional de superacoplamento.

6.2.2 Resposta em frequência

Na configuração considerada, a resposta de transmissão em frequência do sistema, obtida em simulação, é apresentada na Fig. 12. O resultado é comparado com a forma analítica obtida a partir da Eq. 6.13. Constata-se que as curvas são correspondentes, indicando que o comportamento do sistema simulado representa com precisão o modelo analítico. Essa correspondência valida tanto o modelo teórico quanto os parâmetros de entrada utilizados na simulação. Adicionalmente, observa-se que os vales de transmissão se alinham aos comprimentos de onda ressonantes, o que confirma a eficácia do microrressonador em reter energia nas frequências projetadas.

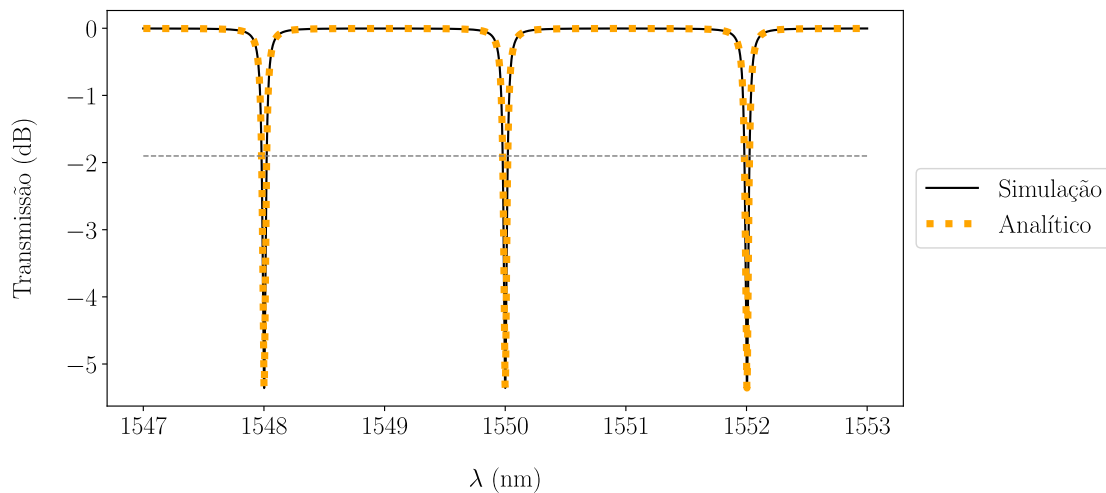


Figura 12 – Espectro de transmissão do MRR implementado obtido por simulação e por expressão analítica. A linha cinza tracejada indica o nível correspondente ao FWHM do espectro.

A partir do espectro de transmissão obtido em simulação, foram medidos os parâmetros de desempenho do MRR, com o objetivo de caracterizá-lo. Os resultados, apresentados na Tab. 3, são comparados com aqueles obtidos a partir das equações analíticas descritas na Seção 5.1.

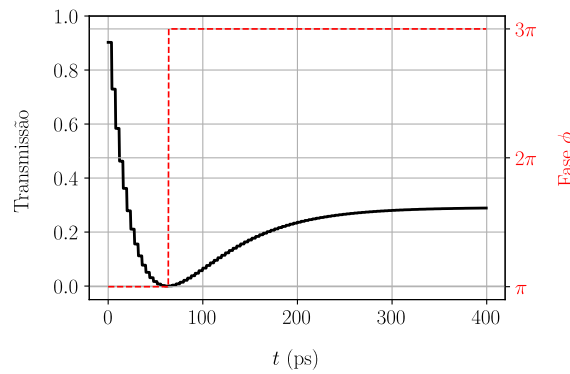
Parâmetro	Simulação	Analítico	ΔE_r (%)
FSR	2,00 nm	2,00 nm	0
FWHM	42,44 pm	42,45 pm	0,02
Q	$3,65 \times 10^4$	$3,65 \times 10^4$	0
Finesse	47,12	47,14	0,04

Tabela 3 – Tabela comparativa entre os parâmetros de desempenho do MRR obtidos a partir da simulação e das equações analíticas.

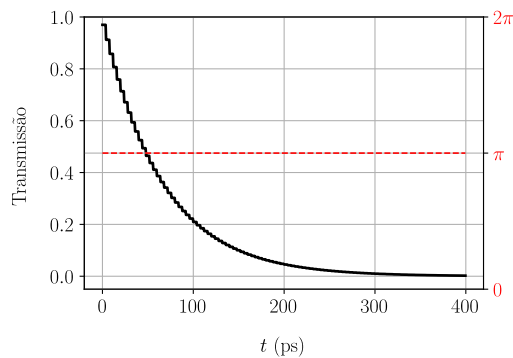
Considerando os erros relativos pouco significativos, observa-se que os valores obtidos nas simulações estão em boa concordância com os cálculos analíticos, o que confirma a precisão do modelo teórico utilizado e valida a confiabilidade do simulador para a análise do comportamento do microrressonador em anel.

6.2.3 Resposta ao degrau

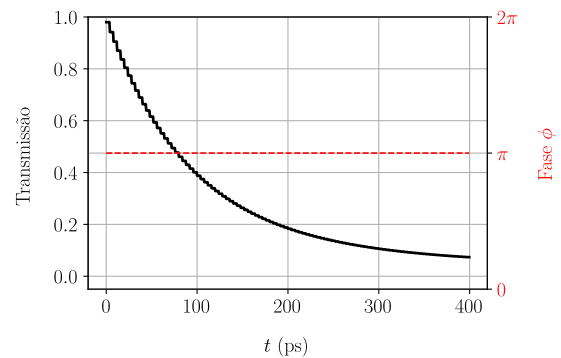
A resposta temporal do MRR implementado a uma entrada degrau unitário com comprimento de onda de 1550 nm (ressonante) é apresentada na Fig. 13. Além disso, na mesma figura, estão exibidas as respostas do mesmo sistema para diferentes coeficientes de transmissão, com o objetivo de analisar o comportamento do MRR nos três regimes de operação distintos.



(a) $r = 0,95$ (super)



(b) $r = 0,98$ (crítico)



(c) $r = 0,99$ (sub)

Figura 13 – Respostas ao degrau (curva preta) e diferença de fase (curva tracejada vermelha) entre campos incidente e transmitido no MRR para diferentes regimes de acoplamento.

Inicialmente, observa-se que, nas três situações, a resposta temporal apresenta uma curva em degraus, com um passo de 4 ps, o que é consistente com o tempo de um ciclo no anel definido para o circuito durante a etapa de design. O sistema apresenta, nesse sentido, uma rede de atraso temporal.

Além disso, como o acoplador direcional não possui uma dimensão física efetiva e está diretamente conectado à fonte, não há atraso inicial no sistema. Assim, logo que o sinal de entrada é aplicado, é possível verificar, no instante $t = 0$ s, um sinal de resposta na porta de passagem, cuja potência transmitida será a fornecida pela fonte, menos a porção acoplada ao anel.

No regime de superacoplamento, após cada ciclo realizado pela luz no ressonador, o campo do sinal que sai do anel interfere com o campo do sinal de entrada na região de acoplamento, gerando o sinal de saída do sistema. Na Fig. 13a, observa-se que ocorre interferência destrutiva nessa situação, à medida que o campo transmitido resultante na porta de passagem diminui progressivamente. Dessa forma, entende-se que a magnitude do campo propagando-se no anel aumenta até atingir um valor suficiente para suprimir completamente o campo transmitido. Esse comportamento foi simulado, e os resultados indicam que são necessárias 15 ciclos consecutivos no anel para que isso ocorra.

A partir desse momento, ocorre um desequilíbrio no mecanismo de interferência, pois o campo circulante no anel se torna tão intenso que passa a dominar na região de acoplamento, provocando um aumento gradual do sinal de saída à medida que parte da potência circulante é transferida para a porta de passagem. Esse processo continua até que a potência transmitida, somada às perdas por atenuação no anel, se iguale à potência do sinal acoplado, momento em que o sistema atinge o regime estacionário.

É possível notar que, nas outras duas condições de acoplamento, ocorre um mecanismo semelhante. No entanto, no acoplamento crítico (cf. Fig. 13b), o regime estacionário é atingido quando o campo transmitido pelo anel interfere destrutivamente com o campo de entrada, extinguindo o sinal na porta de passagem. Já no subacoplamento (cf. Fig. 13c), o regime estacionário ocorre antes que o sinal de saída seja completamente extinto. Assim, nessas condições, não há *undershoot* durante o regime transiente, ocorrendo apenas o decaimento exponencial da potência transmitida.

Os níveis de potência transmitida no regime estacionário podem ser encontrados a partir da aplicação numérica da Eq. 5.3.

Dentro de um esquema de reservatório, os mecanismos de interferência discutidos seriam responsáveis por determinar os pesos de conexão entre os neurônios internos da rede neural recorrente, considerando os demais fenômenos de atenuação e acoplamento envolvidos no funcionamento do microrressonador.

6.3 Aplicação do circuito em RC

O reservatório proposto é implementado a partir do circuito MRR na configuração *all-pass* da Fig. 11. Devido às características temporais do ressonador, a arquitetura utiliza o conceito de multiplexação por divisão de tempo, discutido na Seção 4.2.

Para demonstrar a capacidade do circuito de funcionar como um reservatório, propõe-se a resolução de uma operação XOR- n , que verifica a relação booleana entre um bit presente e um bit n posições anteriores na sequência temporal (cf. Fig. 14). Assim, dada uma sequência de bits $u[k]$ de entrada, o resultado esperado da operação final $t[k]$ é

$$t[k] = u[k] \oplus u[k - 1] \quad (6.14)$$

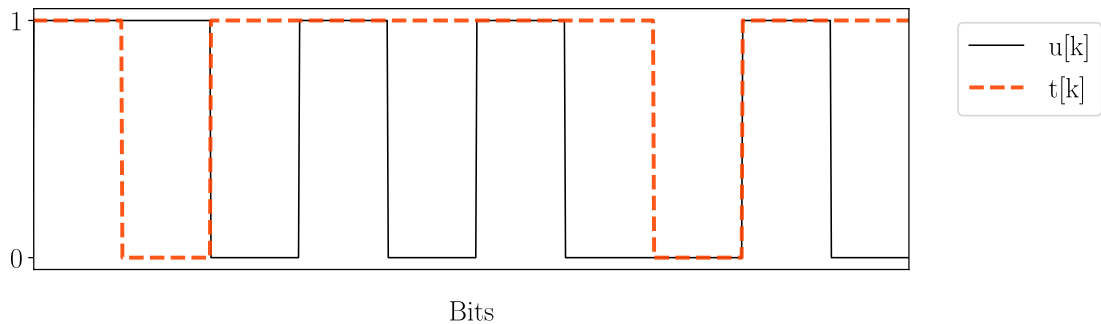


Figura 14 – Resultado de operação XOR-1, realizada em dois bits subsequentes da sequência de entrada (curva preta). Os elementos assumem apenas valores lógicos binários.

Para as análises, considera-se $n = 1$, solucionando-se, portanto, a operação XOR-1, realizada entre dois bits consecutivos da sequência de entrada.

Na etapa de pré-processamento, a sequência de N_b bits $u[k]$ é, inicialmente, transformada em um sinal temporal $u(t)$ com uma taxa de bits f_b , a qual determina a separação temporal θ entre cada unidade virtual pela relação:

$$\theta = \frac{1}{N \cdot f_b} \quad (6.15)$$

Em seguida, o sinal de pulsos quadrados $u(t)$ é submetido à aplicação de uma máscara vetorial M . A dimensão $N \times 1$ da máscara é equivalente ao número N de neurônios virtuais do reservatório, e seus elementos assumem apenas o valor unitário, a fim de não introduzir qualquer viés nos dados de entrada. Esse procedimento corresponde a uma modulação de $u(t)$ por um trem de pulsos unitários, de forma que a sequência modulada pela máscara deve apresentar uma taxa de bits f_{bm} N vezes maior que f_b .

Seguindo esse procedimento, são gerados conjuntos de dados de treinamento e de teste de mesmo tamanho para a resolução do problema de *machine learning*. Devido à capacidade de paralelização do ambiente de simulação em Photontorch, os dois conjuntos são processados simultaneamente em um tensor empilhado. No simulador, os sinais são amostrados com uma frequência f_s , sendo aplicado em tempo contínuo à camada de entrada da arquitetura de RC.

O estado de cada neurônio, para cada uma das entradas binárias da sequência $u[k]$, é obtido amostrando o sinal de saída do circuito fotônico a uma taxa igual a f_{bm} . A amostragem, nesse caso, é equivalente a uma medida da potência do sinal de saída do microrressonador em anel pelo fotodetector. As leituras de intensidade são organizadas em uma matriz de dados X de dimensão $N_b \times N$, de forma que os estados das unidades virtuais podem ser entendidos como os atributos das amostras sequenciais.

O treinamento dos pesos da camada de saída do RC proposto é realizado de maneira *offline*, assim que o sinal de saída é integralmente recuperado na porta de passagem do MRR. Portanto, seja o vetor Y , de dimensão $N_b \times 1$, composto pelos valores esperados \hat{y}_i de cada elemento da sequência. Nessa etapa, determinam-se os elementos do vetor de pesos W_o , de dimensão $N \times 1$, que minimizam o custo quadrático para todos os bits (cf. Eq. 2.8).

Para evitar que a camada de leitura se ajuste excessivamente aos dados de treinamento, levando à condição de sobreaprendizado, utiliza-se regularização (cf. Eq.2.4) por meio de regressão *ridge*, segundo a qual os pesos otimizados são determinados por:

$$W_o^* = \arg \min_{W_o \in \mathbb{R}^N} \|XW_o - Y\|^2 + \|\lambda_{reg}W_o\|^2 \quad (6.16)$$

O coeficiente de regularização λ_{reg} pode ser otimizado utilizando outros conjuntos de treinamento e teste. Entretanto, por simplicidade, para todas as simulações, foi adotado o valor $\lambda_{reg} = 1 \times 10^{-4}$.

A etapa de treinamento foi realizada com o auxílio de ferramentas da biblioteca de Python *scikit-learn* 1.5, desenvolvida de forma *open-source* para facilitar aplicações de análise de dados e *machine learning*.

O desempenho do RC na resolução da operação XOR-n foi avaliado utilizando como métrica a taxa de erro de bits (BER), definida como a razão entre a quantidade de bits preditos de forma errada e o número total de bits N_b da sequência $u[k]$. Nesse sentido, aplicou-se uma função de ativação limiar na camada de leitura, que transforma os resultados da combinação linear XW_o^* em um valor lógico binário, seja 0 ou 1.

6.3.1 Performance

Na Fig. 15, apresenta-se a BER percentual em função do número de unidades virtuais no âmbito da resolução da operação XOR-1. Para cada máscara simulada, utilizou-se dois conjuntos de $N_b = 1000$ bits para treinamento e teste, totalizando um conjunto de dados de 2000 bits. O sinal de entrada foi amostrado à $f_s = 500$ GHz e modulado com uma taxa de bits $f_{bm} = 250$ Gbps. O comprimento de onda da fonte luminosa de entrada foi estabelecido a $\lambda_0 = 1550$ nm.

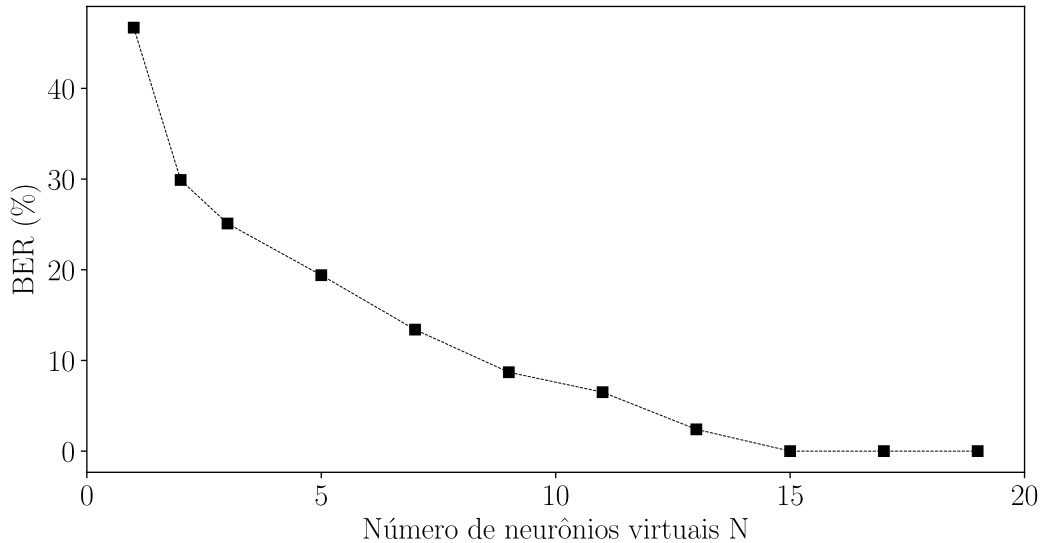


Figura 15 – Taxa percentual de erro de bits em função da quantidade unidades virtuais para operação XOR em dois bits subsequentes

Com uma arquitetura de reservatório contendo 15 neurônios virtuais, é possível obter operações XOR-1 sem erros de resultado. Assumindo que a escala de tempo do sistema τ , definida na Seção 4.2, é equivalente ao tempo $T_R \approx 4$ ps de um ciclo do sinal no anel, observa-se que o número N de unidades virtuais deve ser suficientemente grande para que a duração T de um bit da sequência, dado uma taxa f_{bm} fixa, fosse muito maior que T_R .

Em aplicações práticas, sistemas de RC têm demonstrado desempenhos satisfatórios, alcançando valores de BER na ordem de 10^{-4} [28, 54]. Os valores de BER nula obtidos em simulação refletem as condições ideais de funcionamento, pois não se considera a presença de ruído, efeitos indesejados ou desalinhamentos nos componentes do circuito. De qualquer maneira, esses resultados demonstram a viabilidade teórica de utilizar um MRR integrado como reservatório na configuração *all-pass*, operando em regime linear.

Com base nos resultados anteriores de variação de neurônios, considera-se para o modelo RC fotônico, utilizado neste projeto, uma quantidade de $N = 20$ unidades virtuais com $N_b = 1000$ bits de treinamento e de teste.

Na Fig. 16, tem-se a BER percentual do modelo implementado em função da taxa de bits. A taxa de amostragem utilizada foi definida como $f_s = 10f_b$ e manteve-se o comprimento de onda ressonante.

Os resultados obtidos indicam que o modelo RC fotônico apresenta boa performance, com uma taxa de bits de entrada na ordem de Gbps. De acordo com a Eq. 6.15, uma taxa de erro nula é alcançada para o intervalo de $\theta \in [4, 80]$ ps.

Ao analisar o comportamento da curva $r = 0,95$ na Fig. 13, observa-se que, para um

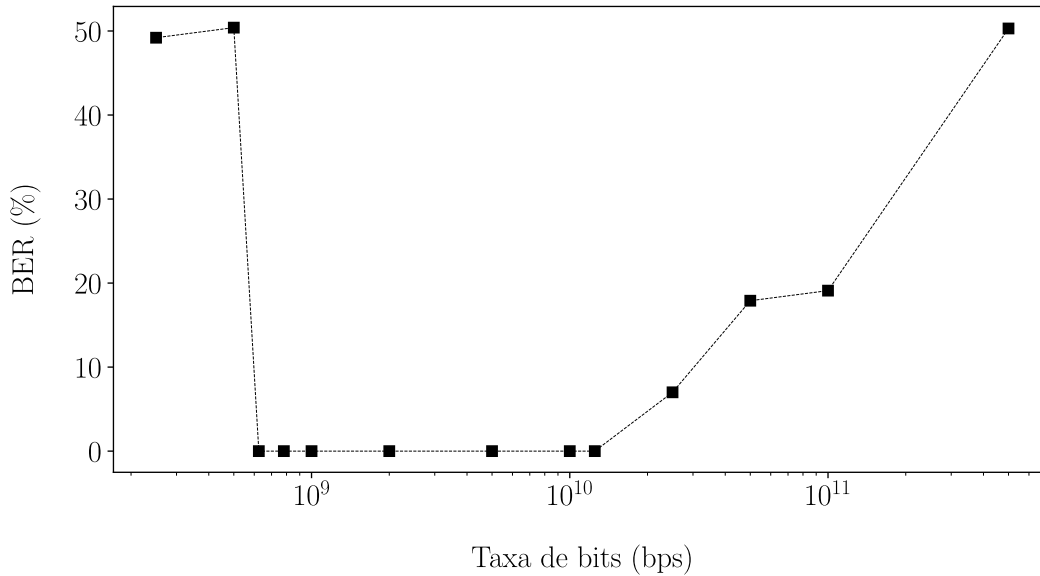


Figura 16 – Taxa percentual de erro de bits em função da taxa de bits para operação XOR em dois bits subsequentes

pulso modulado com duração superior a 80 ps, a resposta do MRR ao pulso se aproxima do regime estacionário antes da entrada de outro pulso no sistema. Isso corrobora a fundamentação teórica apresentada na Seção 4.2, que enfatiza a necessidade de que o reservatório opere, principalmente, no regime transiente. Dessa forma, é necessário modular a sequência de entrada com uma taxa mínima f_{bm} de 12,5 Gbps durante a aplicação da máscara.

Além disso, observa-se que, para valores de θ muito menores que 4 ps, o desempenho do sistema é comprometido. Isso sugere que θ deve ser próximo de T_R . Contudo, os resultados obtidos são ligeiramente diferentes das considerações empíricas da Seção 4.2, na qual se indicou que θ deveria ser menor que T_R .

As dimensões do MRR podem ser reduzidas, o que resulta em uma diminuição de T_R e do intervalo θ , exigindo um aumento na taxa de bits. Esse aumento pode não ser viável para alguns dispositivos geradores de sinal disponíveis no mercado. Nesse contexto, explorar as não-linearidades do microrressonador, como realizado por [29], pode possibilitar um sistema mais compacto e com processamento em alta velocidade.

Durante as simulações de taxa de bits, foi identificada uma restrição em um dos métodos da versão atual da biblioteca Photontorch, que dificultou o processamento de alguns valores de f_b . Por esse motivo, a distribuição de pontos na Fig. 16 não é uniforme. Embora o problema possa ser eventualmente solucionado pelo usuário, devido à característica *open-source* do código, essa limitação não se mostrou fundamental para as análises deste projeto.

6.3.2 Resposta dinâmica do reservatório

Adotando uma taxa de bits f_b de 12,5 Gbps, obteve-se um reservatório com uma resposta dinâmica apresentada na Fig. 17 para um sinal de entrada em $\lambda_0 = 1550$ nm.

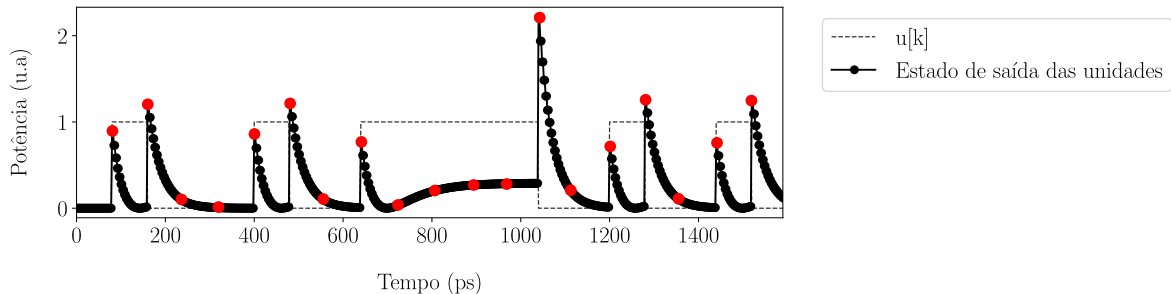


Figura 17 – Resposta dinâmica do circuito fotônico implementado para uma entrada sequencial modulada. Em vermelho, destaca-se o estado de um neurônio ilustrativo ao longo do tempo.

Durante a propagação de um bit 1, que corresponde à entrada de um pulso retangular no sistema, observa-se que a resposta é a mesma obtida para o degrau unitário. Com o término do pulso, se anula a interferência destrutiva na região de acoplamento entre as ondas que se propagam no anel e as ondas transmitidas para a porta de passagem do MRR. Assim, a energia acumulada começa a ser gradualmente transmitida para a saída, evidenciada pelos picos de potência no final dos pulsos. Essa propriedade assegura o armazenamento de estados passados, fundamental para o bom desempenho do reservatório. Além disso, o decaimento exponencial do sinal transmitido é essencial para garantir a memória desvanecente do sistema.

Com relação ao esquema de RC, a Fig. 17 destaca os valores de potência de cada um dos 20 neurônios virtuais, representados por marcadores circulares pretos, que são mensurados pelo fotodetector a partir da resposta do circuito à entrada temporal binária $u(t)$. Nessa mesma figura, os estados temporais de um neurônio virtual são representados por marcadores vermelhos ao longo do tempo. Observa-se que o mesmo neurônio exibe diferentes níveis de potência à medida que a dinâmica transiente do microrressonador varia em resposta às entradas moduladas. Esse comportamento possibilita que os estados temporalmente variantes sejam utilizados para realizar a operação XOR-1, ao serem combinados, em software, com seus pesos associados.

Assim, na Fig. 18, apresenta-se a solução da operação XOR-1 pela camada de leitura após o treinamento dos pesos com os estados das unidades virtuais da Fig. 17.

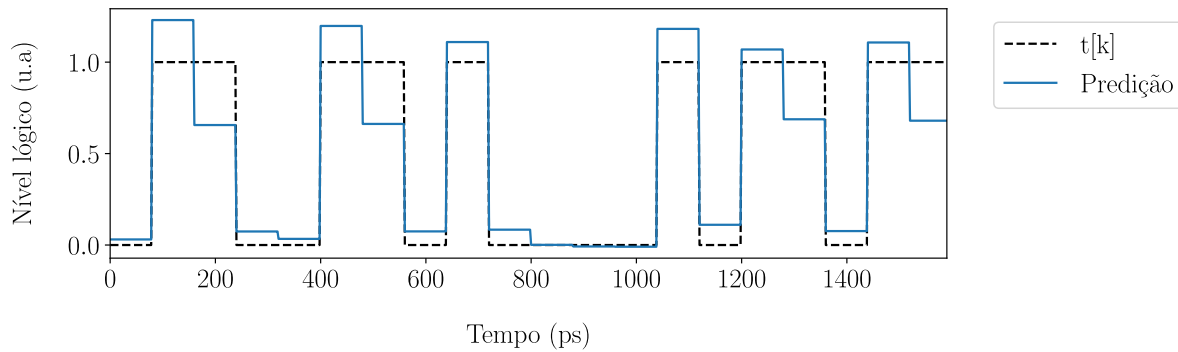


Figura 18 – Resposta preditiva do RC antes da aplicação do limiar em 0.5 e solução esperada $t[k]$ para operação XOR em dois bits consecutivos.

6.3.3 Performance em função do comprimento de onda

Nas simulações da seção anterior, a fonte luminosa utilizada manteve um comprimento de onda correspondente ao ressonante do MRR. Entretanto, conforme previsto pela Eq. 5.3, alterações no comprimento de onda se propagando no anel provocam comportamentos distintos devido à dependência da resposta de transmissão em relação à fase.

Na Fig. 19, observa-se que diferentes comprimentos de onda também resultam em respostas distintas ao degrau unitário no domínio do tempo.

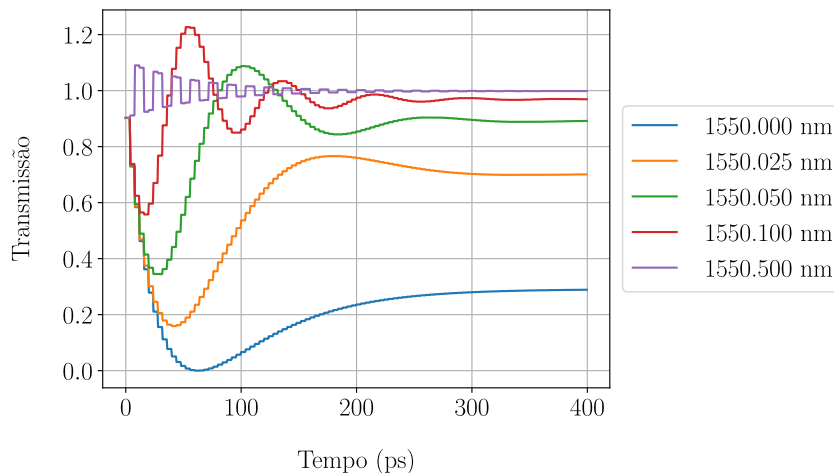


Figura 19 – Respostas ao degrau para um MRR com $r = 0.95$ sujeito a diferentes comprimentos de onda propagantes.

No geral, o deslocamento em comprimento de onda perturba a estabilidade do sistema e degrada o armazenamento de energia circulante no anel, pois a transmissão tende a se aproximar de um valor unitário em regime estacionário.

Assim, a Fig. 20 avalia o comportamento do reservatório em relação a diferentes comprimentos de onda. Nessa simulação, o reservatório projetado foi treinado com 1000

bits para os comprimentos de onda: (a) 1550 nm, (b) 1550.125 nm, (c) 1551 nm e (d) 1551.875 nm. Em seguida, testou-se seu desempenho para diversos comprimentos de onda dentro do intervalo da FSR do sistema, utilizando $N_b = 1000$ bits. O sistema foi amostrado a 500 GHz, e manteve-se a taxa de bits modulados a 250 GHz. Os resultados da BER foram recortados em 10^{-3} .

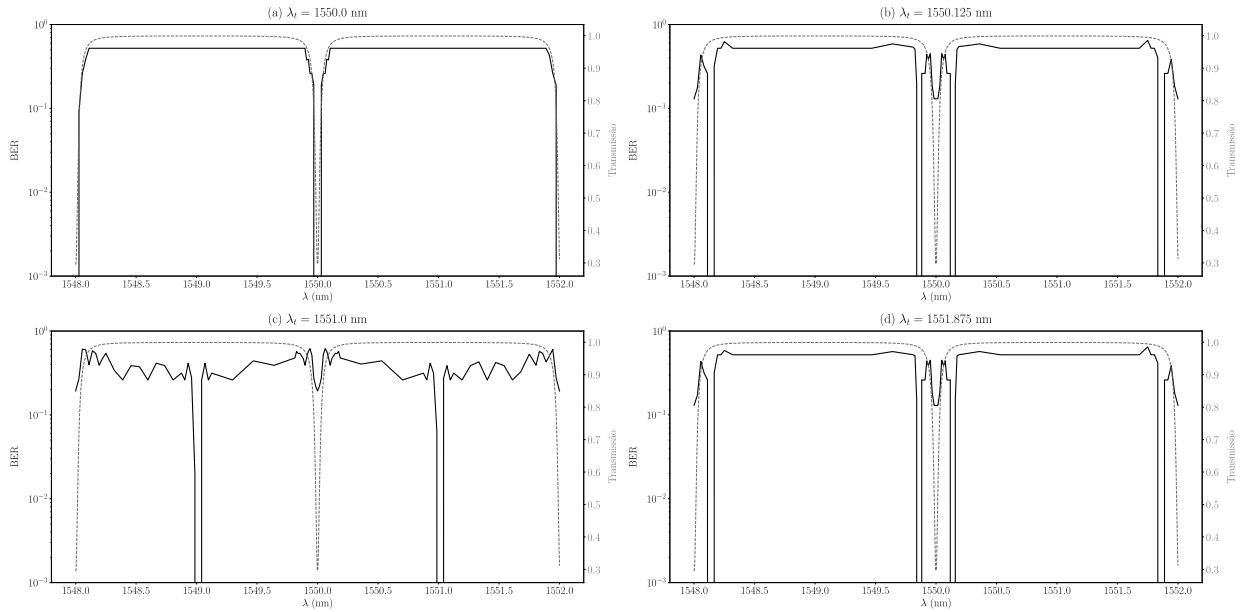


Figura 20 – Taxa de erro de bits para diferentes comprimentos de onda de teste. Em cada uma das quatro situações, o reservatório foi treinado em diferentes comprimentos de onda. A resposta de transmissão é igualmente apresentada para fins comparativos.

Observa-se que o sistema apresenta um comportamento simétrico no espectro, de modo que os pesos ajustados para um determinado comprimento de onda proporcionam desempenho equivalente para comprimentos de onda complementares. No caso do reservatório treinado para 1550 nm, por exemplo, verifica-se que todas as frequências ressonantes resultam em uma taxa de erro de bits nula. Nesse contexto, a complementaridade está associada a comprimentos de onda com a mesma potência transmitida em regime estacionário.

Para os comprimentos de onda nos quais o reservatório foi treinado, o sistema mantém-se robusto em faixas de algumas centenas de picômetros, conforme apresentado na Tabela 4. Define-se como intervalo de robustez os valores para os quais a BER apresenta valores menores que 10^{-3} .

Nas situações (b), (c) e (d), observa-se que o desempenho do reservatório no comprimento de onda ressonante, embora não satisfatório, é um pouco superior ao dos demais comprimentos de onda. Esse resultado evidencia a importância da propriedade ressonante do MRR para garantir uma riqueza de estados no reservatório, proporcionando

Situação	Intervalo de robustez (nm)
(a)	1550,0 ± 0,7
(b)	1550,1 ± 0,4
(c)	1551,0 ± 0,6
(d)	1551,9 ± 0,6

Tabela 4 – Intervalos de robustez para os diferentes comprimentos de onda de treino.

uma melhor conectividade entre os neurônios virtuais, devido ao maior armazenamento de energia circulante no anel nesse comprimento de onda.

6.4 Impacto da variação de temperatura

A variação de temperatura afeta o MRR, alterando tanto seu índice de refração, devido ao efeito termo-óptico, quanto suas dimensões físicas, devido à dilatação térmica. O resultado desses fenômenos é um deslocamento no comprimento de onda ressonante, que pode ser determinado por [70]:

$$\frac{\partial \lambda_0}{\partial T} = \lambda_0(T) \cdot \left[\frac{\left(\frac{\partial n_{eff}}{\partial T} \right) + n_{eff}(\lambda, T) \left(\frac{\partial L}{\partial T} \right) \left(\frac{1}{L} \right)}{n_g(\lambda, T)} \right] \quad (6.17)$$

O coeficiente termo-óptico do silício é $1,86 \times 10^{-4} K^{-1}$ para a banda C, enquanto a variação devido à dilatação térmica depende de um coeficiente aproximadamente 200 vezes menor ($3,57 \times 10^{-6} K^{-1}$). Dessa forma, assume-se que a variação dimensional não altera significativamente a robustez do sistema e, portanto, é desconsiderada na análise.

Embora a Eq. 6.17 seja amplamente utilizada para prever a dependência temporal de ressonadores em anel, algumas aproximações práticas são necessárias para sua aplicação efetiva. Primeiramente, percebe-se que o cálculo dos índices n_{eff} e n_g depende do comprimento de onda para o qual a ressonância será deslocada, enquanto o cálculo do próprio deslocamento depende dos valores previamente determinados desses índices. A interação simultânea entre as variáveis exige que o cálculo da variação do comprimento de onda seja realizado assumindo que os índices se mantêm constantes, com valores previamente estimados em λ_0 (cf. Tab. 2).

Na prática, o deslocamento do comprimento de onda ressonante indica, principalmente, uma variação nos parâmetros de design do MRR utilizados no projeto do circuito. Nesse contexto, como mostrado na Fig. 21a, para a análise em *software* do impacto da variação de temperatura em um comprimento de onda que se propaga no reservatório (linha tracejada), seria necessário implementar diferentes circuitos projetados para as novas condições de ressonância. Esse método corresponderia a definir um ponto de operação nas curvas de transmissão de cada circuito com base no comprimento de onda em análise.

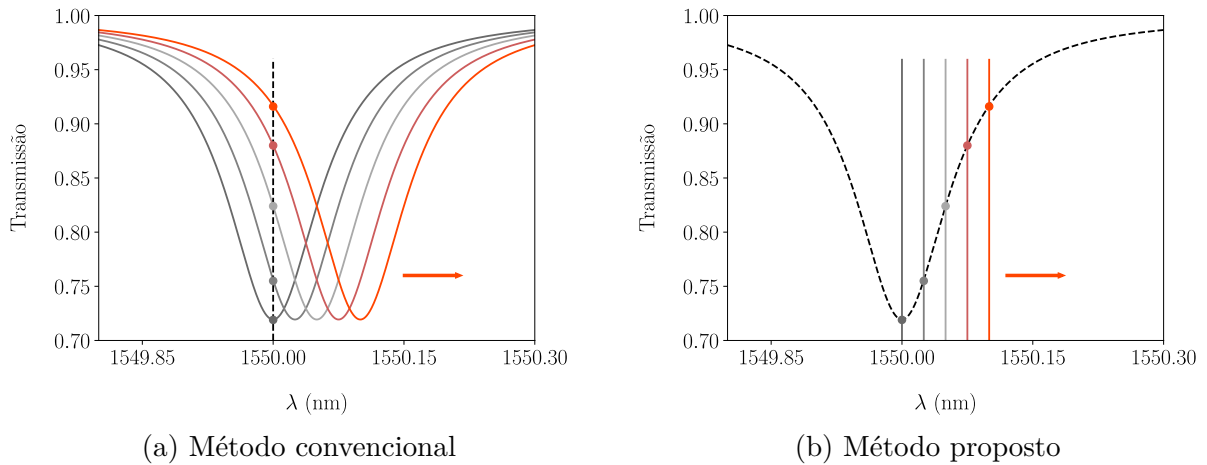


Figura 21 – Comparação entre dois métodos de deslocamento de comprimentos de onda

Para simplificar a simulação, em vez de variar os comprimentos de onda ressonantes, o método proposto neste projeto mantém o mesmo circuito, projetado para o comprimento de onda desejado para análise, e amostra sua resposta de transmissão para diferentes comprimentos de onda, conforme mostrado na Fig.21b. O deslocamento dos comprimentos de onda que se propagam no reservatório é, então, determinado pela variação de temperatura, conforme a Eq.6.18.

$$\lambda = \lambda_0 + \frac{\lambda_0}{n_g} \cdot \frac{\partial n_{eff}}{\partial T} \cdot (T - T_0) \quad (6.18)$$

Além de facilitar a implementação em código, o método de variação de temperatura, baseado na mudança nos comprimentos de onda circulando no sistema, se beneficia da paralelização disponível em Photontorch para vários comprimentos de onda, o que acelera o tempo de simulação.

Vale ressaltar que o índice em λ_0 na Eq. 6.18 não se refere necessariamente ao comprimento de onda ressonante, mas, mais precisamente, ao comprimento de onda que se propaga nas condições iniciais da simulação. Contudo, para a análise realizada, λ_0 será 1550 nm. A temperatura ambiente inicial considerada é $T_0 = 26$ °C.

Baseado nessa proposição metodológica, avalia-se a robustez do reservatório proposto. O RC é treinado com 1000 bits nas condições iniciais (λ_0, T_0) para a operação XOR-1 e, então, sua performance é testada com sequências distintas de 1000 bits para cada uma das temperaturas simuladas, como forma de simultaneamente avaliar a robustez a diferentes conjuntos de dados. As taxas de bits e de amostragem foram configuradas da mesma forma que na Seção 6.3.3. Os resultados de desempenho são apresentados na Fig. 22. Embora o sistema demonstre robustez frente a diferentes sequências de teste, sua performance é significativamente impactada pela variação de temperatura. Observa-se que ele mantém um intervalo de robustez de $26,0 \pm 0,4$ °C, dentro do qual a taxa de erros

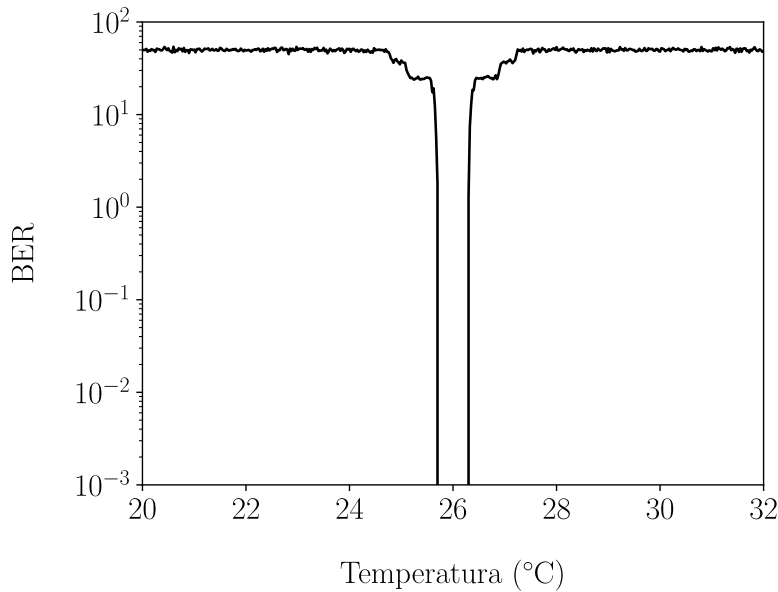


Figura 22 – Taxa de erro de bits para operação XOR-1 em reservatório exposto à variação de temperatura.

é considerada nula. Fora desse intervalo, a BER varia entre 30-50%, refletindo apenas a probabilidade de acertos em um sistema binário.

Conforme discutido na Seção 6.3.3, o comportamento do sistema apresenta simetria e é centralizado na ressonância. No entanto, os resultados obtidos anteriormente indicam que o intervalo de robustez pode variar de acordo com o comprimento de onda para o qual o reservatório foi treinado, uma vez que o desempenho do sistema é influenciado pela resposta de transmissão do MRR.

Para aprofundar essa análise, a robustez foi igualmente avaliada em comprimentos de onda em uma faixa de 100 pm a partir do comprimento de onda ressonante. Os valores centrais dos comprimentos de onda e as larguras dos intervalos de robustez foram determinados com base em resultados como os da Fig. 22 e estão apresentados na Fig. 23.

Em alguns comprimentos de onda, observa-se que a BER pode se tornar menor que o limiar de 10^{-3} em dois ou mais intervalos. Esse comportamento é explicado pela presença de comprimentos de onda complementares, como ilustrado na Fig. 20, que podem ampliar o intervalo de robustez ou introduzir partições adicionais na faixa de comprimentos de onda analisada.

O maior intervalo foi identificado para o comprimento de onda $\lambda_t = 1550,057$ nm, que apresentou robustez em $25,3 \pm 0,9$ °C, o que configura um aumento de 125% em largura quando comparado com o resultado obtido para o reservatório treinado em seu comprimento de onda ressonante. Apesar de a largura dos intervalos não apresentar um padrão evidente, observa-se que a temperatura central se desloca gradativamente da

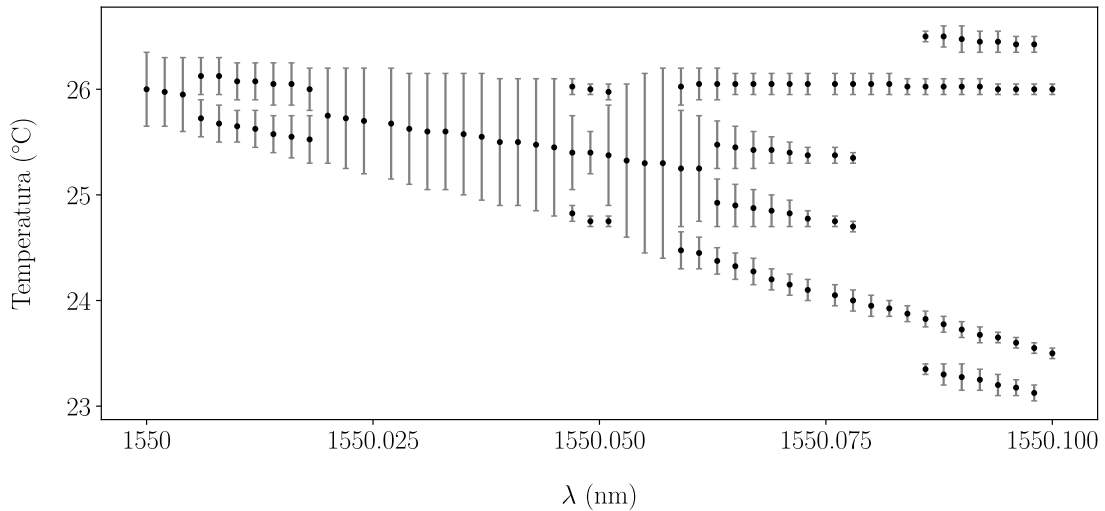


Figura 23 – Intervalos de robustez em temperatura para diferentes comprimentos de onda de treinamento. Os pontos marcados indicam a temperatura central do intervalo e as barras, sua largura.

temperatura inicial $T_0 = 26$ °C à medida que o comprimento de onda de treinamento aumenta.

A partir de 1550,059 nm, os intervalos de robustez tornam-se progressivamente insignificantes e tendem a se concentrar em torno da temperatura ambiente inicial, onde a camada de leitura foi treinada, e no ponto em que a variação de temperatura gera um comprimento de onda complementar ao de treinamento. O desempenho reduzido do reservatório nessa situação deve-se à fração de potência transmitida estar muito próxima da unidade, indicando que o MRR não armazena corretamente os estados passados do sistema.

6.4.1 Condições de acoplamento

Realizou-se uma última análise com o objetivo de verificar o desempenho do sistema frente à variação de temperatura em diferentes condições de acoplamento. Para isso, utilizou-se como circuito base o MRR projetado anteriormente, com a modificação do coeficiente de acoplamento cruzado do acoplador direcional.

Para cada valor de k , o reservatório foi treinado para o comprimento de onda ressonante, e a curva de desempenho foi obtida com uma varredura de temperatura (cf. Fig. 24), utilizando $f_s = 500$ GHz e $f_{bm} = 250$ Gbps.

Nas condições simuladas, observa-se que, para os regimes de subacoplamento e acoplamento crítico, correspondentes aos casos com $k < 0,2$, o reservatório apresenta desempenho insatisfatório, com BER de até 20% no comprimento de onda para o qual foi treinado. Esse comportamento decorre das respostas dinâmicas do sistema nessas

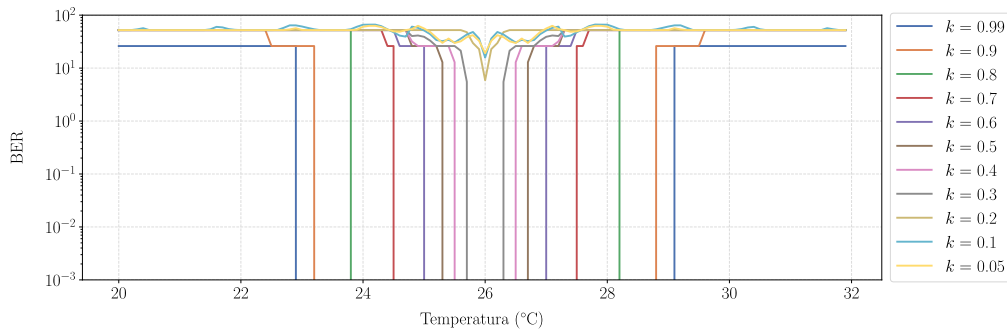


Figura 24 – Curvas de performance do reservatório para diferentes coeficientes de acoplamento cruzado k .

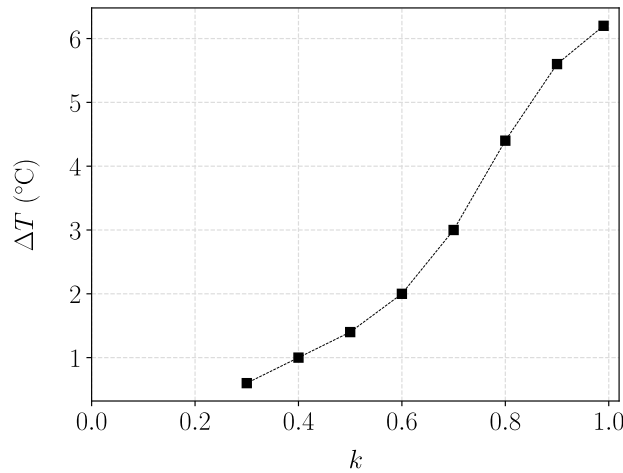


Figura 25 – Intervalos de temperatura nos quais o sistema permanece robusto em função do coeficiente de acoplamento cruzado k . Os resultados obtidos em subacoplamento e acoplamento crítico não foram aproveitados devido ao desempenho insatisfatório nessas condições.

duas condições de acoplamento. À medida que o coeficiente k diminui, menos energia é armazenada no anel, pois o sinal é transmitido diretamente para a porta de passagem do MRR. Nessa situação, realiza-se uma combinação linear na camada de leitura de estados do reservatório, que se torna pouco dependente de estados passados. Com neurônios virtuais vagamente conectados entre si, as características primordiais do RC não são integralmente atendidas, resultando na deterioração do desempenho do sistema na resolução de problemas não lineares.

Recuperando os intervalos de robustez da Fig. 24, produz-se a curva de análise da Fig. 25. Constata-se que o reservatório se torna progressivamente mais robusto, apresentando uma melhoria aproximadamente sigmoide à medida que o coeficiente de transmissão k aumenta, e, conseqüentemente, a energia armazenada no anel durante o funcionamento aumentam.

7 DISCUSSÃO DE RESULTADOS

O sistema RC proposto foi concebido com um reservatório composto por 20 neurônios virtuais e um espaçamento temporal $\theta = 4$ ps, correspondente ao tempo de percurso do anel T_R .

A análise em simulação revelou que o comprimento de onda de treinamento é um parâmetro crítico para a robustez do RC baseado em um MRR em regime de operação linear, influenciando diretamente as condições ideais de performance do sistema. Contudo, mesmo com a escolha ideal de parâmetros para o treinamento da camada de leitura, verificou-se que o sistema apresenta uma robustez limitada frente a flutuações de temperatura, sendo que variações de apenas alguns décimos de grau Celsius podem impactar significativamente os resultados. Dessa forma, considerando uma aplicação prática em larga escala, infere-se que, sob certas condições, como ambientes com grandes variações de temperatura ao longo do dia, mudanças sazonais ou interferências térmicas de outros equipamentos, o reservatório projetado pode não alcançar o desempenho desejado.

Como alternativa, verificou-se que modificações no projeto do circuito podem melhorar significativamente a performance do reservatório. Apesar de o sistema proposto operar no regime de acoplamento necessário, apresenta um coeficiente de transmissão excessivamente elevado, o que compromete a capacidade de armazenamento de energia e, conseqüentemente, a conectividade entre os neurônios virtuais e a memória de estados passados do reservatório. A análise apresentada na última seção sugere que o aumento do coeficiente de acoplamento cruzado pode ser uma solução eficaz para melhorar a robustez de sistemas baseados em MRR.

Além disso, é fundamental que os sistemas RC sejam projetados considerando as condições térmicas de operação. Para mitigar as variações nos comprimentos de onda ressonante, recomenda-se o uso de uma plataforma de controle de temperatura do tipo PID, que pode proporcionar maior estabilidade e robustez ao circuito.

8 CONCLUSÃO

Este trabalho teve como objetivo principal propor um circuito fotônico capaz de operar como um *reservoir computing* e avaliar, em simulação, sua robustez diante de variações de temperatura. Os resultados obtidos indicaram que o sistema, composto por um microrressonador em anel na configuração *all-pass* com raio de $53 \mu\text{m}$, nas condições de acoplamento inicialmente propostas e operando exclusivamente no regime linear, não conseguiu manter uma performance adequada em um intervalo de temperaturas razoável para aplicações práticas. Durante o treinamento para realizar uma operação XOR em uma sequência de bits propagados no comprimento de onda ressonante, o reservatório demonstrou robustez em uma faixa de apenas $0,8 \text{ }^\circ\text{C}$, centralizada na temperatura inicial de $26 \text{ }^\circ\text{C}$. Ao deslocar o comprimento de onda de treinamento em 57 pm , a robustez foi ampliada para um intervalo 2,25 vezes maior, embora assimétrico em relação à temperatura inicial e ainda limitado para aplicações de uso geral. Nesse contexto, o estudo também evidenciou a importância de um trabalho de engenharia cuidadoso no projeto de reservatórios fotônicos, já que melhorias na robustez foram observadas com a escolha adequada dos coeficientes de acoplamento.

A metodologia de análise empregada neste projeto, voltada a um estudo demonstrativo, restringiu-se a modelos fotônicos ideais dentro do formalismo proposto. Apesar disso, a pesquisa apresenta contribuições relevantes para a validação dos efeitos da variação de temperatura na performance de reservatórios fotônicos, além de oferecer procedimentos metodológicos que podem servir como ponto de partida para estudos mais detalhados. Em análises futuras, o refinamento do design será essencial para alcançar simulações que se aproximem do comportamento real dos sistemas. Na prática, os parâmetros de acoplamento dependem de características físicas dos componentes fotônicos não abordadas neste trabalho, como o comprimento da região de acoplamento do acoplador direcional, a distância entre os barramentos do acoplador ou a curvatura desses barramentos. Além disso, ruídos na propagação do sinal, a interação entre detectores/moduladores e outras fontes de atenuação também não foram considerados, destacando a necessidade de modelos mais abrangentes para o desenvolvimento de aplicações práticas.

Além disso, destaca-se que o uso de softwares de simulação mais robustos pode ser necessário em estudos futuros. O pacote Photontorch provou ser uma ferramenta extremamente útil e eficiente para estudos comportamentais de sistemas fotônicos, como o realizado neste trabalho. No entanto, a plataforma apresenta limitações em comparação com outras opções disponíveis no mercado, como Lumerical, COMSOL e Lucedra, particularmente no que diz respeito a ferramentas avançadas de análise e modelagem de componentes.

Este projeto também se apresentou como um suporte introdutório à implemen-

tação de reservatórios fotônicos. Ao longo do estudo, investigou-se o comportamento do microrressonador em anel proposto como reservatório tanto no domínio da frequência quanto no do tempo. Assim, foi possível atingir o objetivo de consolidar um estudo que contribui para uma melhor compreensão do funcionamento do microrressonador dentro do paradigma de *reservoir computing*.

Com relação à extensão para potenciais aplicações práticas, considerando que a computação neuromórfica busca avanços no processamento de dados, pesquisas futuras podem ampliar a avaliação de performance frente a variações térmicas para outros sistemas de reservatórios fotônicos. Essa abordagem permitiria identificar possibilidades gerais de melhoria na robustez e contribuir para o estabelecimento de figuras de mérito, facilitando comparações eficazes entre diferentes arquiteturas de reservatórios.

Espera-se que este estudo amplie o potencial da computação neuromórfica fotônica, fortalecendo seu papel em cenários tecnológicos emergentes.

REFERÊNCIAS

- [1] E. González-Sarmiento, J. Roa-Perez, e L. Ortiz-Ospino, “Big Data and Artificial Intelligence in the Development of Industry 4.0; A Bibliometric Analysis,” *IOP Conference Series: Materials Science and Engineering*, vol. 1154, n. 1, p. 012008,
- [2] T. J. Sejnowski, “The unreasonable effectiveness of deep learning in artificial intelligence,” *Proceedings of the National Academy of Sciences*, vol. 117, n. 48, pp. 30 033–30 038,
- [3] M. M. Waldrop, “More than Moore,” *Nature*, vol. 530, n. 7589, pp. 144–148,
- [4] K. Rupp, “42 Years of Microprocessor Trend Data,” *www.karlrupp.net*,
- [5] R. Schaller, “Moore’s law: past, present and future,” *IEEE Spectrum*, vol. 34, n. 6, pp. 52–59,
- [6] N. Harris, “Extending Moore’s Law with Integrated Photonics (and Packaging),” *IMAPSource Proceedings*, vol. 2023, n. DPC, pp. 1087–1117,
- [7] N. C. Thompson, K. Greenewald, K. Lee, e G. F. Manso, “The Computational Limits of Deep Learning,”
- [8] S. Shankar e A. Reuther, “Trends in energy estimates for computing in ai/machine learning accelerators, supercomputers, and compute-intensive applications,” em *2022 IEEE High Performance Extreme Computing Conference (HPEC)*, 2022, 2022.
- [9] F. Zhu, P. Xu, e J. Zong, “Moore’s Law: The potential, limits, and breakthroughs,” *Applied and Computational Engineering*, vol. 10, pp. 307–315,
- [10] D. Marković, A. Mizrahi, D. Querlioz, e J. Grollier, “Physics for neuromorphic computing,” *Nature Reviews Physics*, vol. 2, pp. 499–510,
- [11] D. Miller e H. Ozaktas, “Limit to the Bit-Rate Capacity of Electrical Interconnects from the Aspect Ratio of the System Architecture,” *Journal of Parallel and Distributed Computing*, vol. 41, n. 1, pp. 42–52,
- [12] B. Jalali e S. Fathpour, “Silicon Photonics,” *Journal of Lightwave Technology*, vol. 24, n. 12, pp. 4600–4615,
- [13] Y. A. Vlasov e S. J. McNab, “Losses in single-mode silicon-on-insulator strip waveguides and bends,” *Optics Express*, vol. 12, n. 8, pp. 1622–1631,

-
- [14] R. Marchetti, V. Vitali, C. Lacava, I. Cristiani, B. Charbonnier, V. Muffato, M. Fournier, e P. Minzioni, “Group-velocity dispersion in SOI-based channel waveguides with reduced-height,” *Optics Express*, vol. 25, n. 9, pp. 9761–9767,
- [15] G. F. R. Chen, J. R. Ong, T. Y. L. Ang, S. T. Lim, C. E. Png, e D. T. H. Tan, “Broadband Silicon-On-Insulator directional couplers using a combination of straight and curved waveguide sections,” *Scientific Reports*, vol. 7, n. 1, p. 7246,
- [16] X. Wang, W. Shi, H. Yun, S. Grist, N. A. F. Jaeger, e L. Chrostowski, “Narrow-band waveguide Bragg gratings on SOI wafers with CMOS-compatible fabrication process,” *Optics Express*, Vol. 20, Issue 14, pp. 15547-15558,
- [17] C. Lacava, M. J. Strain, P. Minzioni, I. Cristiani, e M. Sorel, “Integrated nonlinear Mach Zehnder for 40 Gbit/s all-optical switching,” *Optics Express*, vol. 21, n. 18, pp. 21 587–21 595,
- [18] W. Bogaerts, P. De Heyn, T. Van Vaerenbergh, K. De Vos, S. Kumar Selvaraja, T. Claes, P. Dumon, P. Bienstman, D. Van Thourhout, e R. Baets, “Silicon microring resonators,” *Laser & Photonics Reviews*, vol. 6, n. 1, pp. 47–73,
- [19] D. J. Thomson *et al.*, “50-Gb/s Silicon Optical Modulator,” *IEEE Photonics Technology Letters*, vol. 24, n. 4, pp. 234–236,
- [20] X. Mu, S. Wu, L. Cheng, e H. Y. Fu, “Edge Couplers in Silicon Photonic Integrated Circuits: A Review,” *Applied Sciences*, vol. 10, n. 4, p. 1538,
- [21] Z. Wang *et al.*, “Novel Light Source Integration Approaches for Silicon Photonics,” *Laser & Photonics Reviews*, vol. 11, n. 4, p. 1700063,
- [22] B. J. Shastri, A. N. Tait, T. Ferreira de Lima, W. H. P. Pernice, H. Bhaskaran, C. D. Wright, e P. R. Prucnal, “Photonics for artificial intelligence and neuromorphic computing,” *Nature Photonics*, vol. 15, n. 2, pp. 102–114,
- [23] Y. Shen *et al.*, “Deep Learning with Coherent Nanophotonic Circuits,” *Nature Photonics*, vol. 11, n. 7, pp. 441–446,
- [24] A. Mehrabian, Y. Al-Kabani, V. J. Sorger, e T. El-Ghazawi, “PCNNA: A Photonic Convolutional Neural Network Accelerator,” em *2018 31st IEEE International System-on-Chip Conference (SOCC)*, Set. 2018, Set. 2018.
- [25] J. Feldmann, N. Youngblood, C. D. Wright, H. Bhaskaran, e W. H. P. Pernice, “All-optical spiking neurosynaptic networks with self-learning capabilities,” *Nature*, vol. 569, n. 7755, pp. 208–214,
- [26] G. Van Der Sande, D. Brunner, e M. C. Soriano, “Advances in photonic reservoir computing,” *Nanophotonics*, vol. 6, n. 3, pp. 561–576,

- [27] G. Tanaka, T. Yamane, J. B. Héroux, R. Nakane, N. Kanazawa, S. Takeda, H. Numata, D. Nakano, e A. Hirose, “Recent advances in physical reservoir computing: A review,” *Neural Networks*, vol. 115, pp. 100–123,
- [28] M. Borghi, S. Biasi, e L. Pavesi, “Reservoir computing based on a silicon microring and time multiplexing for binary and analog operations,” *Scientific Reports*, vol. 11, n. 1, p. 15642,
- [29] G. Donati, C. R. Mirasso, M. Mancinelli, L. Pavesi, e A. Argyris, “Microring resonators with external optical feedback for time delay reservoir computing,” *Optics Express*, vol. 30, n. 1, pp. 522–537,
- [30] G. Donati, A. Argyris, M. Mancinelli, C. R. Mirasso, e L. Pavesi, “Time delay reservoir computing with a silicon microring resonator and a fiber-based optical feedback loop,” *Optics Express*, vol. 32, n. 8, pp. 13 419–13 437,
- [31] N. N. Klimov, S. Mittal, M. Berger, e Z. Ahmed, “On-chip silicon waveguide Bragg grating photonic temperature sensor,” *Optics Letters*, vol. 40, n. 17, p. 3934,
- [32] S. Zarei, “Design and analysis of a fiber-optic deep-etched silicon photonic crystal temperature sensor,” *Journal of Electromagnetic Waves and Applications*, vol. 33, n. 2, pp. 226–235,
- [33] C. Hu, Y. Shi, T. Zhou, C. Xu, e A. Zhu, “A Small Size On-chip Temperature Sensor Based on a Microring Resonator,” *Silicon*, vol. 14, n. 11, pp. 5847–5854,
- [34] T. Hastie, R. Tibshirani, e J. Friedman, *The Elements of Statistical Learning*, ser. Springer Series in Statistics, 2009.
- [35] G. James, D. Witten, T. Hastie, R. Tibshirani, e J. Taylor, *An Introduction to Statistical Learning: with Applications in Python*, ser. Springer Texts in Statistics, 2023.
- [36] T. Hastie, R. Tibshirani, e M. Wainwright, *Statistical Learning with Sparsity: The Lasso and Generalizations*, Abr. 2015.
- [37] A. Zhang, Z. C. Lipton, M. Li, e A. J. Smola, *Dive into Deep Learning*, 2023,
- [38] F. Rosenblatt, “The perceptron: A probabilistic model for information storage and organization in the brain.” *Psychological Review*, vol. 65, n. 6, pp. 386–408,
- [39] I. Goodfellow, Y. Bengio, e A. Courville, *Deep Learning*, 2016.
- [40] W. S. McCulloch e W. Pitts, “A logical calculus of the ideas immanent in nervous activity,” *The bulletin of mathematical biophysics*, vol. 5, n. 4, pp. 115–133,

-
- [41] H. Wang, B. Raj, e E. P. Xing, “On the origin of deep learning,” *CoRR*, vol. abs/1702.07800,
- [42] D. E. Rumelhart, G. E. Hinton, e R. J. Williams, “Learning representations by back-propagating errors,” *Nature*, vol. 323, n. 6088, pp. 533–536,
- [43] D. P. Kingma e J. Ba, “Adam: A Method for Stochastic Optimization,” *CoRR*,
- [44] A. Graves, *Supervised Sequence Labelling with Recurrent Neural Networks*, ser. Studies in Computational Intelligence, 2012. Berlin, Heidelberg: Springer, 2012,
- [45] P. J. Werbos, “Generalization of backpropagation with application to a recurrent gas market model,” *Neural Networks*, vol. 1, n. 4, pp. 339–356,
- [46] R. Pascanu, T. Mikolov, e Y. Bengio, “On the difficulty of training Recurrent Neural Networks,”
- [47] S. Hochreiter, “The Vanishing Gradient Problem During Learning Recurrent Neural Nets and Problem Solutions,” *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 06, n. 02, pp. 107–116,
- [48] H. Jaeger, “The “echo state” approach to analysing and training recurrent neural networks,” *GMD-Report 148*, German National Research Institute for Computer Science,
- [49] W. Maass, T. Natschläger, e H. Markram, “Real-Time Computing Without Stable States: A New Framework for Neural Computation Based on Perturbations,” *Neural Computation*, vol. 14, n. 11, pp. 2531–2560,
- [50] T. M. Cover, “Geometrical and Statistical Properties of Systems of Linear Inequalities with Applications in Pattern Recognition,” *IEEE Transactions on Electronic Computers*, vol. EC-14, n. 3, pp. 326–334,
- [51] K. Nakajima e I. Fischer, Eds., *Reservoir Computing: Theory, Physical Implementations, and Applications*, ser. Natural Computing Series, 2021.
- [52] L. Boccatto, A. Lopes, R. Attux, e F. J. Von Zuben, “An echo state network architecture based on volterra filtering and PCA with application to the channel equalization problem,” em *The 2011 International Joint Conference on Neural Networks*, Jul. 2011, Jul. 2011.
- [53] K. Vandoorne, J. Dambre, D. Verstraeten, B. Schrauwen, e P. Bienstman, “Parallel Reservoir Computing Using Optical Amplifiers,” *IEEE Transactions on Neural Networks*, vol. 22, n. 9, pp. 1469–1481,

- [54] K. Vandoorne, P. Mechet, T. Van Vaerenbergh, M. Fiers, G. Morthier, D. Verstraeten, B. Schrauwen, J. Dambre, e P. Bienstman, “Experimental demonstration of reservoir computing on a silicon photonics chip,” *Nature Communications*, vol. 5, n. 1, p. 3541,
- [55] F. Laporte, A. Katumba, J. Dambre, e P. Bienstman, “Numerical demonstration of neuromorphic computing with photonic crystal cavities,” *Optics Express*, vol. 26, n. 7, pp. 7955–7964,
- [56] M. A. A. Fiers, T. Van Vaerenbergh, F. Wyffels, D. Verstraeten, B. Schrauwen, J. Dambre, e P. Bienstman, “Nanophotonic Reservoir Computing With Photonic Crystal Cavities to Generate Periodic Patterns,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, n. 2, pp. 344–355,
- [57] C. Mesaritakis, V. Papataxiarhis, e D. Syvridis, “Micro ring resonators as building blocks for an all-optical high-speed reservoir-computing bit-pattern-recognition system,” *Journal of the Optical Society of America B*, vol. 30, n. 11, p. 3048,
- [58] L. Appeltant, M. Soriano, G. Van Der Sande, J. Danckaert, S. Massar, J. Dambre, B. Schrauwen, C. Mirasso, e I. Fischer, “Information processing using a single dynamical node as complex system,” *Nature Communications*, vol. 2, n. 1, p. 468,
- [59] L. Appeltant, G. Van Der Sande, J. Danckaert, e I. Fischer, “Constructing optimized binary masks for reservoir computing with delay systems,” *Scientific Reports*, vol. 4, n. 1, p. 3629,
- [60] Y. Paquot, F. Duport, A. Smerieri, J. Dambre, B. Schrauwen, M. Haelterman, e S. Massar, “Optoelectronic Reservoir Computing,” *Scientific Reports*, vol. 2, n. 1, p. 287,
- [61] M. Masi, R. Orobtcouk, Guofang Fan, J.-M. Fedeli, e L. Pavesi, “Towards a Realistic Modelling of Ultra-Compact Racetrack Resonators,” *Journal of Lightwave Technology*, p. 5585641,
- [62] J. Heebner, R. Grover, e T. A. Ibrahim, *Optical Microresonators: Theory, Fabrication, and Applications*, 2008.
- [63] J. Heebner, Vincent Wong, A. Schweinsberg, R. Boyd, e D. Jackson, “Optical transmission characteristics of fiber ring resonators,” *IEEE Journal of Quantum Electronics*, vol. 40, n. 6, pp. 726–730,
- [64] F. Laporte, J. Dambre, e P. Bienstman, “Highly parallel simulation and optimization of photonic circuits in time and frequency domain based on the deep-learning framework PyTorch,” *Scientific Reports*, vol. 9, n. 1, p. 5918,

- [65] F. Laporte, “Novel architectures for brain-inspired photonic computers,” dissertation, Ghent University,
- [66] M. Fiers, T. V. Vaerenbergh, K. Caluwaerts, D. V. Ginste, B. Schrauwen, J. Dambre, e P. Bienstman, “Time-domain and frequency-domain modeling of nonlinear optical components at the circuit-level using a node-based approach,” *JOSA B*, vol. 29, n. 5, pp. 896–900,
- [67] F. Laporte, “Photontorch documentation,” <https://docs.photontorch.com>,
- [68] C. D. Salzberg e J. J. Villa, “Infrared Refractive Indexes of Silicon Germanium and Modified Selenium Glass*,” *JOSA*, vol. 47, n. 3, pp. 244–246,
- [69] S. Xiao, M. H. Khan, H. Shen, e M. Qi, “Compact silicon microring resonators with ultra-low propagation loss in the C band,” *Optics Express*, vol. 15, n. 22, pp. 14 467–14 475,
- [70] H. Xu, M. Hafezi, J. Fan, J. M. Taylor, G. F. Strouse, e Z. Ahmed, “Ultra-sensitive chip-based photonic temperature sensor using ring resonator structures,” *Optics Express*, vol. 22, n. 3, pp. 3098–3104,