



UFOP

Universidade Federal
de Ouro Preto

**Universidade Federal de Ouro Preto
Instituto de Ciências Exatas e Aplicadas
Departamento de Computação e Sistemas**

Redes Neurais Artificiais no âmbito da Computação Quântica

Fernanda Bernardes da Silva Melo

TRABALHO DE CONCLUSÃO DE CURSO

**ORIENTAÇÃO:
Dr. Ronan Silva Ferreira**

**Agosto, 2023
João Monlevade–MG**

Fernanda Bernardes da Silva Melo

Redes Neurais Artificiais no âmbito da Computação Quântica

Orientador: Dr. Ronan Silva Ferreira

Monografia apresentada ao curso de Engenharia de Computação do Instituto de Ciências Exatas e Aplicadas, da Universidade Federal de Ouro Preto, como requisito parcial para aprovação na Disciplina “Trabalho de Conclusão de Curso II”.

Universidade Federal de Ouro Preto

João Monlevade

Agosto de 2023

SISBIN - SISTEMA DE BIBLIOTECAS E INFORMAÇÃO

M528r Melo, Fernanda Bernardes da Silva.
Redes neurais artificiais no âmbito da computação quântica.
[manuscrito] / Fernanda Bernardes da Silva Melo. - 2023.
53 f.: il.: color., gráf., tab..

Orientador: Prof. Dr. Ronan Silva Ferreira.
Monografia (Bacharelado). Universidade Federal de Ouro Preto.
Instituto de Ciências Exatas e Aplicadas. Graduação em Engenharia de
Computação .

1. Computação quântica. 2. Inteligência artificial. 3. Modelos
matemáticos. 4. Redes neurais (Computação). I. Ferreira, Ronan Silva. II.
Universidade Federal de Ouro Preto. III. Título.

CDU 004.8

Bibliotecário(a) Responsável: Flavia Reis - CRB6-2431



FOLHA DE APROVAÇÃO

Fernanda Bernardes da Silva Melo

Redes Neurais Artificiais no âmbito da Computação Quântica

Monografia apresentada ao Curso de Engenharia de Computação da Universidade Federal de Ouro Preto como requisito parcial para obtenção do título de Bacharel em Engenharia de Computação

Aprovada em 01 de Setembro de 2023

Membros da banca

Dr - Ronan Silva Ferreira - Orientador - (Universidade Federal de Ouro Preto)
Dr - Alexandre Magno de Souza - (Universidade Federal de Ouro Preto)
Dr - Talles Henrique de Medeiros - (Universidade Federal de Ouro Preto)

Ronan Silva Ferreira, orientador do trabalho, aprovou a versão final e autorizou seu depósito na Biblioteca Digital de Trabalhos de Conclusão de Curso da UFOP em 17/11/2023



Documento assinado eletronicamente por **Ronan Silva Ferreira, PROFESSOR DE MAGISTERIO SUPERIOR**, em 17/11/2023, às 13:05, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **0626102** e o código CRC **AEC9B1F4**.

Este trabalho é dedicado à minha família e aos meus professores.

Agradecimentos

Agradeço à Universidade Federal de Ouro Preto, pela oferta do curso de Engenharia de Computação, pelas diversas oportunidades acadêmicas oferecidas, pela estrutura e oportunidade de capacitação profissional. Agradeço também aos meus prezados professores, que contribuíram diretamente para a minha formação e crescimento durante a graduação, me incentivando a continuar e demonstrando verdadeira devoção ao ato de ensinar.

Agradeço ao meu orientador Ronan, pelo imensurável conhecimento compartilhado durante todos esses anos em que estivemos juntos em disciplinas, iniciações científicas e na execução deste trabalho. Pela dedicação como exímio professor e pela seriedade como leva seu papel como profissional da educação, pela amizade e pelos conselhos.

Por fim, agradeço principalmente à minha família, pelo suporte e pelo incentivo. Especialmente à minha mãe, que sempre foi o meu maior exemplo pessoal, acadêmico e profissional. Graças à ela, me mantive resiliente e encorajada durante todo o curso.

“Nature isn’t classical, dammit, and if you want to make a simulation of nature, you’d better make it quantum mechanical, and by golly it’s a wonderful problem, because it doesn’t look so easy.”

— Richard Feynman (1918 – 1988),
in: Simulating Physics with Computers.

Resumo

As Redes Neurais Artificiais são modelos matemáticos que buscam reproduzir o funcionamento do cérebro humano. É um dos pilares da Inteligência Artificial que possui algoritmos clássicos na literatura, como é o caso do modelo de neurônio artificial Perceptron. Por sua simplicidade de implementação, permite o estudo de uma migração para um modelo quântico. O objetivo deste trabalho é analisar e construir um modelo quântico do Perceptron através da compreensão dos principais conceitos que cercam as duas áreas. Para a análise do poder computacional do algoritmo Perceptron implementado, foi proposto e realizado o experimento de classificação de cores de 8 *bits* utilizando apenas um neurônio Perceptron. Os resultados mostraram a capacidade de predição correta para todos os testes realizados. Além disso, foi evidenciada a capacidade de rompimento da limitação de classificação de dados lineares, imposta pelo modelo clássico, quando utilizado o modelo quântico, de forma com que apenas um neurônio é capaz de reconhecer letras e cores. Essa característica abre espaço para a possibilidade de reconhecimento de padrões ainda mais complexos.

Palavras-chaves: redes neurais artificiais. perceptron. computação quântica.

Abstract

Artificial Neural Networks are mathematical models that seek to reproduce the functioning of the human brain. It is one of the pillars of Artificial Intelligence that has classic algorithms in the literature, such as the Perceptron artificial neuron model. Due to its simplicity of implementation, it allows the study of a migration to a quantum model. The objective of this work is to analyze and build a quantum model of the Perceptron through the understanding of the main concepts that surround the two areas. To analyze the computational power of the implemented Perceptron algorithm, an 8-bit color classification experiment using only one Perceptron neuron was proposed and carried out. The results showed the correct prediction ability for all tests performed. In addition, the ability to break the limitation of linear data classification, imposed by the classical model, when using the quantum model, was evidenced, so that only one neuron is able to recognize letters and colors. This feature makes room for the possibility of recognizing even more complex patterns.

Key-words: artificial neural networks. perceptron. quantum computing.

Lista de ilustrações

Figura 1 – Base computacional.	19
Figura 2 – Analogia entre portas lógicas clássicas e quânticas. (a) Porta lógica clássica: dadas duas entradas binárias, a saída é um valor binário. (b) Porta lógica quântica: dada entrada um estado quântico, a saída é um estado quântico oriundo da atuação da porta sobre o estado de entrada.	23
Figura 3 – Representação da porta lógica quântica X atuando sobre um estado de 2 qubits.	24
Figura 4 – Representação da porta lógica quântica Z atuando sobre um estado de 2 qubits.	25
Figura 5 – Representação da porta lógica quântica Hadamard atuando sobre um estado de 2 qubits.	25
Figura 6 – Clonagem de um bit a partir da porta lógica XOR.	26
Figura 7 – Clonagem de um estado quântico a partir da CNOT.	27
Figura 8 – Operador U_f	30
Figura 9 – Circuito de Deutsch.	30
Figura 10 – Representação geométrica de U_w atuando sobre os estados $ \psi_i\rangle$ e $ \psi_w\rangle$	39
Figura 11 – Possível arranjo para um circuito quântico que representa o Perceptron para $N = 2$	42
Figura 12 – Exemplo de representação dos vetores de entrada e peso.	43
Figura 13 – a. Resultado ideal para o Perceptron, simulado em um processador clássico. b. Resultado da simulação simulada em um processador quântico. Além disso, é explicitado o valor da discrepância média correspondente em relação ao caso ideal.	44
Figura 14 – Simulação para vetores \vec{i} e \vec{w} com 16 elementos.	45
Figura 15 – Exemplo de representação das letras A, B e C através de imagens de 8×8 pixels.	46
Figura 16 – Resultado obtido através da execução do Perceptron para todas as possíveis combinações de letras.	46

Lista de tabelas

Tabela 1 – Tabela de correspondência entre o número de bits e qubits para descrever um mesmo número de estados.	20
---	----

Lista de abreviaturas e siglas

RNA Redes Neurais Artificiais

IA Inteligência Artificial

SF Signal Flip

Lista de símbolos

α	Letra grega minúscula alpha
β	Letra grega minúscula beta
δ	Letra grega minúscula delta
ϕ	Letra grega minúscula phi
θ	Letra grega minúscula theta
ψ	Letra grega minúscula psi
\in	Pertence
\oplus	Disjunção exclusiva
\otimes	Produto tensorial

Sumário

1	INTRODUÇÃO	14
1.1	O problema de pesquisa	15
1.2	Objetivos	15
1.3	Metodologia	16
1.4	Organização do trabalho	16
2	REVISÃO BIBLIOGRÁFICA	18
2.1	Informação Quântica	18
2.1.1	Qubit	18
2.1.2	Emaranhamento	20
2.1.3	Portas lógicas quânticas	23
2.1.3.1	Porta lógica X	24
2.1.3.2	Porta lógica Z	24
2.1.3.3	Porta lógica Hadamard	25
2.1.4	Teorema da Não-Clonagem	26
2.1.5	Paralelismo quântico	28
2.2	Redes Neurais Artificiais	35
2.2.1	Perceptron Clássico	35
2.3	Trabalhos Relacionados	37
3	DESENVOLVIMENTO	38
3.1	Perceptron Quântico	38
3.1.1	Construção do modelo	40
4	RESULTADOS	43
4.0.1	Evidências computacionais	43
4.0.2	Classificando cores de 8 bits	47
5	CONCLUSÃO	50
	REFERÊNCIAS	52

1 Introdução

A Computação Quântica representa um avanço significativo no campo da Engenharia de Computação, destacando-se como uma das áreas mais promissoras e desafiadoras do momento. Em outubro de 2019, a Google anunciou um marco importante ao alcançar a “supremacia quântica” (ARUTE et al., 2019), demonstrando que seu processador quântico Sycamore executou uma tarefa em cerca de 200 segundos, que qualquer supercomputador clássico de última geração, levaria 10.000 anos para executar. Esse feito impressionante impulsiona novas perspectivas e aplicações para a Engenharia de Computação, especialmente em problemas de natureza matemática, como é o caso da Inteligência Artificial (IA).

As Redes Neurais Artificiais (RNA), pilares da Inteligência Artificial, apresentam um conjunto de técnicas com a intenção de mapear o cérebro humano em modelos matemáticos, para que esses modelos sejam capazes de processar informações, tomar decisões e aprender de alguma forma. Neste contexto, o Perceptron, proposto por Rosenblatt em 1958 (ROSENBLATT, 1958), representa um marco seminal na literatura das Redes Neurais Artificiais. Esse neurônio artificial simples e pioneiro demonstrou a capacidade de aprendizado por meio de um único neurônio, estabelecendo os fundamentos para arquiteturas mais complexas de redes neurais.

A interseção da Computação Quântica com Redes Neurais Artificiais abre caminho para explorar novas dimensões no aprendizado de máquina. Partindo do Perceptron clássico, que opera com decisões binárias, abordamos sua versão quântica considerando princípios da mecânica quântica, possibilitando cálculos paralelos mais eficientes.

O trabalho está compreendido em três partes principais. Primeiro, exploramos os fundamentos da Computação Quântica, incluindo qubits, emaranhamento e princípio de superposição, portas lógicas quânticas, impossibilidade de cópia e paralelismo. Segundo, analisamos detalhadamente o Perceptron clássico, abordando arquitetura, funcionamento e limitações. Essa base viabiliza a apresentação e entendimento do Perceptron quântico na terceira parte, explorando sua capacidade de aprendizado e classificação, com simulações demonstrando seu potencial além da classificação linear.

O objetivo é investigar a convergência entre a Computação Quântica e as Redes Neurais Artificiais, destacando as possibilidades e desafios dessa união. Ao compreender as capacidades do Perceptron quântico, construímos um modelo através do entendimento dos principais conceitos que cercam as duas áreas. A partir desse modelo, foi possível propor e realizar o experimento de classificação de cores de 8 bits, o que evidenciou uma grande vantagem em se utilizar a versão quântica do Perceptron: o modelo foi capaz de classificar corretamente todas as combinações de cores com apenas um neurônio, o que explicita a

perda da necessidade de linearidade dos dados, limitação intrínseca no modelo clássico.

Com esse resultado, este trabalho abre precedentes para o estudo de modelos mais complexos de RNA no âmbito da computação quântica, contribuindo com uma densa revisão bibliográfica e com a demonstração da capacidade de classificação de um modelo para o Perceptron quântico.

1.1 O problema de pesquisa

Um aspecto crucial da atualidade relacionado à IA e suas limitações é a quantidade massiva de dados que os algoritmos de IA precisam processar. Esse volume de dados pode tornar os cálculos extremamente complexos, mesmo para algoritmos aparentemente simples. No entanto, há uma esperança de avanço por meio da aplicação da computação quântica.

A computação quântica é uma área emergente da ciência da computação que explora os princípios da mecânica quântica para processar informações de maneira fundamentalmente diferente da computação clássica. Uma das propriedades-chave da computação quântica é a superposição, onde bits quânticos, conhecidos como qubits, podem existir em múltiplos estados simultaneamente. Isso possibilita uma abordagem altamente paralela para o processamento de informações, o que poderia levar a um ganho exponencial no poder de processamento em comparação com a computação clássica.

Dessa maneira, a computação quântica oferece a perspectiva de superar os desafios enfrentados pelos algoritmos de IA devido à quantidade massiva de dados. Através da superposição e manipulação de qubits, surge o potencial de realizar cálculos complexos de maneira mais eficiente e rápida, abrindo novas possibilidades para a resolução de problemas que atualmente são considerados intratáveis devido à sua complexidade. Essa promessa levanta a esperança de que os algoritmos de IA possam alcançar níveis de desempenho ainda mais impressionantes, à medida que o processamento de dados seja otimizado por essa abordagem.

1.2 Objetivos

O objetivo deste estudo é a compreensão de uma rede neural baseada nas versões clássica e quântica do Perceptron, investigando aspectos da computação quântica. Entre os objetivos específicos que podem ser citados estão:

- Estudar, conhecer e documentar os princípios da Informação Quântica.
- Pesquisar abordagens a respeito da interseção entre a Computação Quântica e a IA.

- Construir e validar um modelo de Perceptron Quântico.

1.3 Metodologia

- Estudar, conhecer e documentar os princípios da Informação Quântica.** Foi realizada uma ampla revisão da literatura, abordando os principais autores, livros e artigos a respeito do tema. Os tópicos englobados neste estudo foram: (i) *qubit*, a unidade básica da informação quântica, sua representação e principais aspectos; (ii) *emaranhamento e o princípio de superposição*, analisando como os qubits podem ser trabalhados de forma conjunta e a capacidade de se extrair informações deles sem necessariamente conhecer o valor de cada unidade individualmente; (iii) *portas lógicas quânticas*, que atuam sobre os qubits capacitando sua manipulação e a construção de circuitos quânticos; (iv) *teorema da não-clonagem*, abordando a incapacidade de clonagem de estados quânticos e a promessa em áreas como a criptografia; (v) *paralelismo quântico*, evidenciando a capacidade intrínseca de a computação quântica processar dados de forma paralela.
- Pesquisar abordagens a respeito da interseção entre a Computação Quântica e a IA.** Foi utilizado como principal meio de pesquisa o Google Scholar¹, utilizando termos chave como *Quantum Computing*, *Machine Learning in Quantum Computing* e *Quantum Perceptron*.
- Construir e validar um modelo de Perceptron Quântico.** Para esse objetivo, foi escolhido o modelo proposto em “*An artificial neuron implemented on an actual quantum processor*” (ARUTE et al., 2019). Foi realizada um estudo minucioso a respeito do algoritmo proposto, desde sua construção até os resultados obtidos. Além disso foi analisada também a tese de mestrado “*Quantum algorithm for the implementation of a perceptron*” (BON, 2019), que baseou-se e realizou experimentos utilizando esse modelo do Perceptron. Por fim, propusemos e realizamos um experimento com o objetivo de classificar cores de 8 bits utilizando o modelo.

1.4 Organização do trabalho

O restante deste trabalho é organizado como se segue. O Capítulo 2 apresenta uma ampla revisão bibliográfica que discorre sobre os principais fundamentos da Informação Quântica, percorrendo desde o funcionamento dos qubits até o paralelismo quântico. No Capítulo 3 é apresentado e esmiuçado o modelo de Perceptron utilizado como base

¹ <https://scholar.google.com/>

para este estudo. Por fim, no capítulo Capítulo 4 são apresentados os resultados obtidos através da execução desse algoritmo quântico. Esse capítulo é dividido em duas partes principais: (i) a evidência do funcionamento do algoritmo, testado através de processadores quânticos e simulados pelos autores do artigo e tese usados como referência, bem como a análise e descrição dos resultados alcançados (seção 4.0.1); (ii) a simulação matemática-computacional da classificação de cores em 8 bits utilizando apenas um neurônio Perceptron Quântico (seção 4.0.2).

2 Revisão bibliográfica

Nas seções que seguem, são destacados os fundamentos da computação quântica. Será visto que graças ao princípio de superposição, propriedade que faz com que a computação quântica tenha um grau de paralelismo natural, há uma grande promessa envolvendo a inteligência artificial e o aprendizado de máquina. Além disso, a impossibilidade de cópia e a capacidade de troca de informações permitida através do emaranhamento abrem portas para uma criptografia mais segura e eficaz. Na seção 2.3 são evidenciados trabalhos relacionados a este.

2.1 Informação Quântica

A informação quântica é uma forma de informação que é codificada e manipulada usando os princípios da mecânica quântica. Esse tipo de informação difere da clássica, que é a forma de informação que estamos mais familiarizados, de várias maneiras. Por exemplo, a informação quântica pode ser superposta, o que significa que pode estar em dois estados diferentes ao mesmo tempo. Isso permite que a informação quântica seja processada de maneiras que a informação clássica não pode.

Portanto, acredita-se que a informação quântica tem o potencial de revolucionar muitas áreas da tecnologia, incluindo computação, comunicação e segurança. Os computadores quânticos podem ser exponencialmente mais rápidos que os computadores clássicos, e podem ser usados para resolver problemas que são atualmente impossíveis de resolver com computadores clássicos. Os sistemas de comunicação quânticos podem ser usados para transmitir informações de forma segura, mesmo que sejam interceptadas. E a criptografia quântica pode ser usada para proteger informações de acesso não autorizado.

Nesta seção serão abordados os tópicos básicos e necessários para a compreensão dos fundamentos da Informação Quântica e suas características, além de abordar quais são realmente os benefícios em termos de processamento ao se utilizar essa tecnologia.

2.1.1 Qubit

Um qubit pode ser entendido como *um sistema quântico de dois níveis* (COHEN-TANNOUDJI; LALOE; DIU, 2019) com um grau de liberdade, que pode assumir um de dois estados, $|0\rangle$ ou $|1\rangle$ ¹. Na prática, pode-se tomar um sistema desses como sendo a polarização de um fóton ou o *spin* de uma partícula, por exemplo. Quando o valor de um qubit é medido, é encontrado exatamente o que se espera de um bit (clássico): 0 ou

¹ Dirac notation.

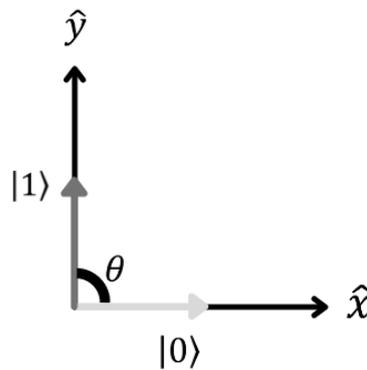
1. A vantagem da versão quântica como unidade lógica de computação é o princípio de superposição que governa esses sistemas, que permite com que sua unidade básica esteja em mais de um estado simultaneamente, e não apenas 0 ou 1. Nesse sentido, o qubit é a unidade básica de computação, não o bit. Um qubit pode estar em uma infinidade de estados de superposição durante os cálculos computacionais. É uma espécie de paralelismo intrínseco. Um estado qualquer $|\psi\rangle$ pode ser escrito na forma:

$$\begin{aligned} |\psi\rangle &= a|0\rangle + b|1\rangle \\ &= a \begin{pmatrix} 1 \\ 0 \end{pmatrix} + b \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ &= \begin{pmatrix} a \\ b \end{pmatrix} \end{aligned} \quad (2.1)$$

sendo a e b números complexos², em que $|a|^2 + |b|^2 = 1$ e na segunda linha é usada a notação matricial, dos vetores coluna. Por essa razão, pode ser dito que a probabilidade de se obter o estado $|0\rangle$ é de $|a|^2$, enquanto para o estado $|1\rangle$ a probabilidade é igual a $|b|^2$. Aqui, $|0\rangle$ e $|1\rangle$ são vetores unitários que compõem uma base ortonormal: não há projeção de um sobre o outro. Dito de outra forma, se um qubit é medido em $|0\rangle$, então não foi medido em $|1\rangle$. É óbvio, mas ajuda a enxergar a representação matemática a seguir.

É possível representar o estado $|\psi\rangle = |0\rangle$ como um vetor na direção $\theta = 0$, no sentido $+\hat{x}$ (horizontal, para direita), enquanto $|\psi\rangle = |1\rangle$ na direção $\theta = \pi/2$, no sentido $+\hat{y}$ (vertical, para cima). A Figura 1 mostra essa representação.

Figura 1 – Base computacional.



Fonte: Produzida pelos autores.

Dessa forma, se essa base for rotacionada em torno de um ângulo α , o efeito será

$$\begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \longrightarrow \begin{pmatrix} \cos \alpha \\ -\sin \alpha \end{pmatrix}, \begin{pmatrix} \sin \alpha \\ \cos \alpha \end{pmatrix} \quad (2.2)$$

² Um número complexo possui a forma $z = x + iy$, em que $i = \sqrt{-1}$ é um operador de rotação no plano complexo Im .vs. Re.

Observe que girando em torno de $\alpha = \pi/2$, obtém-se $\{-|1\rangle, |0\rangle\}$. Isso nada mais é que a base com componentes trocadas: $|0\rangle$ passa à vertical pra baixo, enquanto $|1\rangle$ horizontal pra direita. Dito de outra forma, todo o espectro de possibilidades pode ser mapeado em $\theta = [0, \pi]$ e $\alpha = [0, \pi/2]$. Disso decorre que $\theta = 2\alpha$ e, de forma geral, $|\psi\rangle$ pode ser escrito como:

$$|\psi\rangle = \alpha \begin{pmatrix} \cos \theta/2 \\ -\sin \theta/2 \end{pmatrix} + \beta \begin{pmatrix} \sin \theta/2 \\ \cos \theta/2 \end{pmatrix}. \quad (2.3)$$

2.1.2 Emaranhamento

Os qubits podem ser trabalhados de maneira conjunta. Neste ponto, a vantagem em se trabalhar com a versão quântica do bit é notória. Por exemplo, com dois bits, é possível descrever quatro estados: 00, 01, 10 e 11. Como o qubit pode estar em um estado de superposição, a mesma descrição desses quatro estados pode ser feita com apenas um qubit. A relação qubit *versus* bit para descrever um mesmo número de estados é descrita pela expressão $bits = 2^{qubits}$ (ver Tabela 1).

qubit	bit
1	2
2	4
4	16
8	256
10	1.024
n	2^n

Tabela 1 – Tabela de correspondência entre o número de bits e qubits para descrever um mesmo número de estados.

A combinação de dois qubits é descrita pelo produto tensorial de dois estados, $|\psi\rangle \otimes |\phi\rangle = |\psi\phi\rangle$. Para exemplificar isso, é possível usar o produto tensorial de uma das duas bases $\theta = 0$ com ela mesma:

$$|00\rangle = \begin{bmatrix} \alpha_{11} = 1 \\ \alpha_{21} = 0 \end{bmatrix} \otimes \begin{bmatrix} \beta_{11} = 1 \\ \beta_{21} = 0 \end{bmatrix} = \begin{bmatrix} \alpha_{11}\beta_{11} = 1 \\ \alpha_{11}\beta_{21} = 0 \\ \alpha_{21}\beta_{11} = 0 \\ \alpha_{21}\beta_{21} = 0 \end{bmatrix}. \quad (2.4)$$

Portanto, a base do novo espaço vetorial combinado será $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$ ou

$$\left(\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \right). \quad (2.5)$$

Com efeito, um estado qualquer deste espaço combinado pode ser descrito na forma

$$|\psi\rangle = a|00\rangle + b|01\rangle + c|10\rangle + d|11\rangle, \quad (2.6)$$

sendo ψ aqui uma superposição dos estados lógicos da base, em que a, b, c, d são amplitudes de probabilidade que cada um desses estados possa ocorrer. Dito de outra forma, deve-se garantir que $a^2 + b^2 + c^2 + d^2 = 1$.

Até aqui é possível afirmar que dois qubits estejam *combinados*, mas não necessariamente *emaranhados*. A propriedade do emaranhamento irá depender da relação lógica $ad \neq bc$ entre as amplitudes de probabilidade “puras”, a e d , e as “mistas” b e c . Para ilustrar o emaranhamento, imagine duas moedas quânticas, $M1$ e $M2$, que naturalmente possuem dois níveis: $|C\rangle$ e $|K\rangle$, respectivamente, *Cara* e *Coroa*. O estado combinado dessas duas moedas quânticas é dado por $|M1M2\rangle = m|CC\rangle + n|CK\rangle + p|KC\rangle + q|KK\rangle$. Para que essas moedas estejam emaranhadas, é necessário ter $mq \neq np$. Por exemplo, $m = n = 1/2$, $p = 1/\sqrt{2}$ e $q = 0$. Imagine agora que, após serem emaranhadas, a moeda $M1$ seja entregue a Alice e $M2$ entregue a Bob, que estão separados geograficamente. Portanto, o primeiro qubit em $|M1M2\rangle$ está com Alice e o segundo com Bob. Do ponto de vista matemático, e de Alice, temos:

$$\frac{1}{\sqrt{2}}|C_A\rangle \left(\frac{1}{\sqrt{2}}|C_B\rangle + \frac{1}{\sqrt{2}}|K_B\rangle \right) + \frac{1}{\sqrt{2}}|K_A\rangle (1|C_B\rangle + 0|K_B\rangle), \quad (2.7)$$

enquanto do ponto de vista de Bob

$$\frac{1}{\sqrt{2}}|C_B\rangle \left(\frac{1}{\sqrt{2}}|C_A\rangle + \frac{1}{\sqrt{2}}|K_A\rangle \right) + \frac{1}{\sqrt{2}}|K_B\rangle (1|C_A\rangle + 0|K_A\rangle). \quad (2.8)$$

Observe que Alice - equação (2.7) - pode encontrar sua moeda no estado $|C_A\rangle$ ou $|K_A\rangle$ com 50%, uma vez que as amplitudes de probabilidades são iguais a $1/\sqrt{2}$ para ambos os casos. Entretanto, esses resultados para a moeda de Alice acarretam em probabilidades diferentes para que Bob encontre sua moeda em um dos estados $|C_B\rangle$ ou $|K_B\rangle$. Se Alice encontrar, por exemplo, o estado $|C_A\rangle$, Bob pode encontrar sua moeda no estado $|C_B\rangle$ ou $|K_B\rangle$, ambos com 50% de chance. É o que mostra o primeiro termo da equação (2.7) - Lembre-se que se Alice encontra $|C_A\rangle$, o segundo termo vai a zero:

$$|C_A\rangle \longrightarrow \left(\frac{1}{\sqrt{2}}|C_B\rangle + \frac{1}{\sqrt{2}}|K_B\rangle \right). \quad (2.9)$$

Por outro lado, se Alice encontra $|K_A\rangle$, Bob só pode estar com sua moeda no estado $|C_B\rangle$, uma vez que nesta situação a probabilidade de Bob encontrar sua moeda em $|K_B\rangle$ é nula,

$$|K_A\rangle \longrightarrow |C_B\rangle. \quad (2.10)$$

O emaranhamento, importante pilar da computação quântica, é uma consequência direta do princípio de superposição. Com ele, é possível fazer previsões sobre um determinado sistema “ A ” a partir de um sistema “ B ”, mesmo se eles estiverem separados a distâncias muito grandes, o que permite que informações sejam transmitidas a partir de qubits.

É possível obter um estado emaranhado a partir da base padrão, usando o circuito de Bell (JAEGGER, 2004), que retorna o que é conhecida como a *base de Bell*. Esse circuito é composto por uma porta lógica Hadamard (2.1.3.3) e uma porta CNOT (2.1.4), de maneira que a entrada é uma entrada padrão $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$, e a saída é do tipo:

$$Bell(|00\rangle) = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle), \quad (2.11)$$

$$Bell(|01\rangle) = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle), \quad (2.12)$$

$$Bell(|10\rangle) = \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle), \quad (2.13)$$

$$Bell(|11\rangle) = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle). \quad (2.14)$$

Para verificar que realmente tratam-se de estados emaranhados, considere:

$$|a\rangle = \alpha_1 |0\rangle + \beta_1 |1\rangle, \quad (2.15)$$

$$|b\rangle = \alpha_2 |0\rangle + \beta_2 |1\rangle. \quad (2.16)$$

Em que α e β são amplitudes de probabilidade. Fazendo o produto tensorial

$$|a\rangle \otimes |b\rangle = (\alpha_1 |0\rangle + \beta_1 |1\rangle) \otimes (\alpha_2 |0\rangle + \beta_2 |1\rangle) \quad (2.17)$$

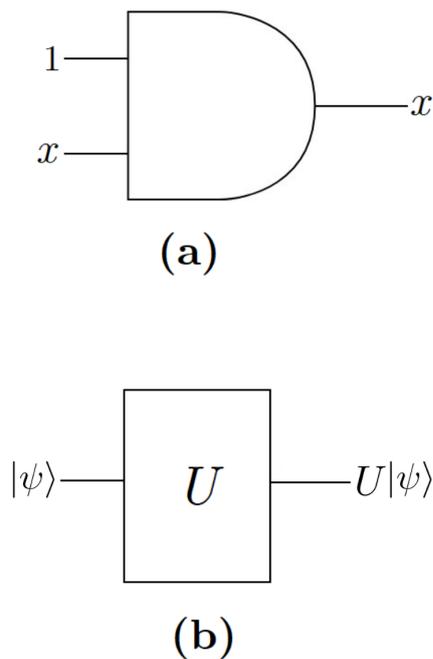
$$= \alpha_1 \alpha_2 |00\rangle + \alpha_1 \beta_2 |01\rangle + \alpha_2 \beta_1 |10\rangle + \beta_1 \beta_2 |11\rangle \quad (2.18)$$

Vamos comparar este resultado com o estado de Bell $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$. Observe que considerando $\alpha_1 \beta_2 = 0$, necessariamente, α_1 é 0 ou β_2 é 0, o que mostra que é impossível encontrar coeficientes tais que o estado final possa ser representado por um produto tensorial, o que é característico do estado emaranhado. Além disso, é possível perceber que as previsões se confirmam, uma vez que em $\frac{1}{\sqrt{2}}(|0\rangle_A |1\rangle_B + |1\rangle_A |0\rangle_B)$ se o qubit A for medido e o resultado encontrado for “0”, o resultado encontrado por B deve ser “1” e vice-versa.

2.1.3 Portas lógicas quânticas

A computação quântica, assim como a computação clássica, fundamenta-se em um modelo de processamento de informações que envolve circuitos compostos por portas lógicas. Enquanto as portas lógicas na computação clássica operam em bits discretos, alterando-os de acordo com regras específicas, as portas quânticas atuam em qubits e até mesmo estados previamente preparados, manipulando-os de acordo com os princípios da mecânica quântica. Essas portas lógicas quânticas podem realizar operações complexas e poderosas, incluindo superposição, emaranhamento e inversão de fase, que permitem a realização de cálculos de forma simultânea em várias possibilidades. A Figura 2 mostra como uma porta lógica quântica atua em comparação com uma porta lógica clássica.

Figura 2 – Analogia entre portas lógicas clássicas e quânticas. (a) Porta lógica clássica: dadas duas entradas binárias, a saída é um valor binário. (b) Porta lógica quântica: dada entrada um estado quântico, a saída é um estado quântico oriundo da atuação da porta sobre o estado de entrada.



Fonte: Produzida pelos autores.

No mundo clássico, existem algumas portas fundamentais, como a NOT, AND e OR. Na quântica, não é diferente. A seguir, serão apresentadas algumas das portas lógicas fundamentais que compõem os circuitos quânticos (BARENCO et al., 1995) e que serão relevantes para o entendimento deste trabalho.

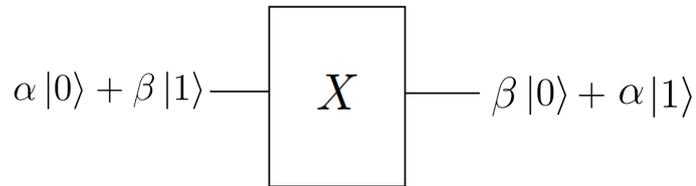
2.1.3.1 Porta lógica X

Também conhecida como porta lógica X de Pauli (NIELSEN; CHUANG, 2010), é uma porta de negação, análoga à porta NOT no mundo clássico, sendo capaz de inverter o valor de um qubit, ou seja, troca $|0\rangle$ por $|1\rangle$ e vice-versa. Matematicamente, a porta lógica X é representada pela seguinte matriz:

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad (2.19)$$

A Figura 3 mostra como a porta X atua sobre o estado $\alpha|0\rangle + \beta|1\rangle$.

Figura 3 – Representação da porta lógica quântica X atuando sobre um estado de 2 qubits.



Fonte: Produzida pelos autores.

O que ocorre neste caso, é exatamente a inversão de valor de ambos os qubits. Matematicamente, o que este operador unitário faz é:

$$X|0\rangle = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} = |1\rangle \quad (2.20)$$

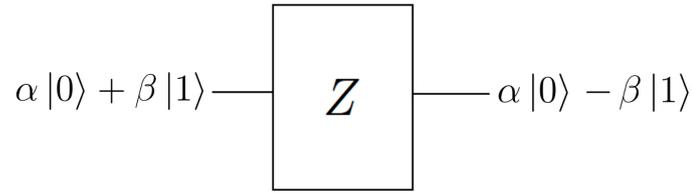
$$X|1\rangle = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} = |0\rangle \quad (2.21)$$

2.1.3.2 Porta lógica Z

A porta lógica quântica Z é uma porta de fase, que aplica uma mudança de fase global ao estado do qubit. Também é conhecida como porta lógica Z de Pauli (NIELSEN; CHUANG, 2010) e não altera os valores dos qubits $|0\rangle$ e $|1\rangle$, mas aplica uma fase global de -1 ao qubit $|1\rangle$. Matematicamente, a porta lógica Z é representada pela seguinte matriz:

$$Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad (2.22)$$

A Figura 4 mostra como a porta Z atua sobre o estado $\alpha|0\rangle + \beta|1\rangle$.

Figura 4 – Representação da porta lógica quântica Z atuando sobre um estado de 2 qubits.

Fonte: Produzida pelos autores.

O que ocorre neste caso, é a mudança de fase do estado $|1\rangle$ para $-|1\rangle$. Matematicamente, o que este operador unitário faz é:

$$Z |0\rangle = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} = |0\rangle \quad (2.23)$$

$$Z |1\rangle = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = - \begin{bmatrix} 0 \\ 1 \end{bmatrix} = -|1\rangle \quad (2.24)$$

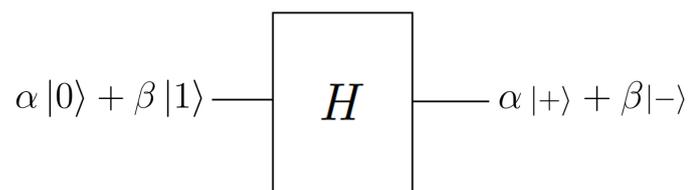
2.1.3.3 Porta lógica Hadamard

A porta Hadamard é uma das portas quânticas mais importantes e é citada diversas vezes ao longo deste trabalho. Ela realiza uma transformação que leva os estados de base $\{|0\rangle, |1\rangle\}$ para superposições desses estados. É especialmente útil para criar emaranhamento e gerar distribuições de probabilidade uniformes (DJORDJEVIC, 2012). Matematicamente, a porta Hadamard é representada pela seguinte matriz:

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (2.25)$$

A Figura 5 mostra como a porta Hadamard atua sobre o estado $\alpha |0\rangle + \beta |1\rangle$.

Figura 5 – Representação da porta lógica quântica Hadamard atuando sobre um estado de 2 qubits.



Fonte: Produzida pelos autores.

O que ocorre é a transformação do estado $|0\rangle$ em $|+\rangle$ ³ e de $|1\rangle$ para $|-\rangle$ ⁴, ou seja, dois estados em que existe igual probabilidade de o qubit ser medido como $|0\rangle$ ou $|1\rangle$ e $|0\rangle$ ou $-|1\rangle$. Matematicamente, o que este operador unitário faz é:

$$H|0\rangle = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix} = |+\rangle \quad (2.26)$$

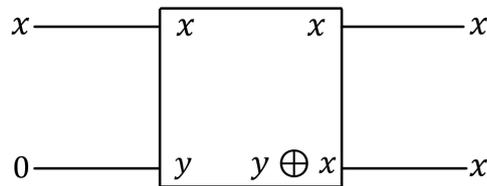
$$H|1\rangle = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = - \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix} = |-\rangle \quad (2.27)$$

2.1.4 Teorema da Não-Clonagem

Por meio do emaranhamento quântico, vimos que é possível transmitir informações entre partículas separadas a grandes distâncias. Mas essa informação pode ser clonada? Nos computadores clássicos, podemos facilmente copiar informações de um lugar para outro, basta utilizarmos um CD ou disco rígido. Porém, nos computadores quânticos, é impossível a clonagem de informação.

Podemos demonstrar a cópia de um bit em um sistema clássico a partir da porta lógica “XOR”, ou, “ou exclusivo”.

Figura 6 – Clonagem de um bit a partir da porta lógica XOR.



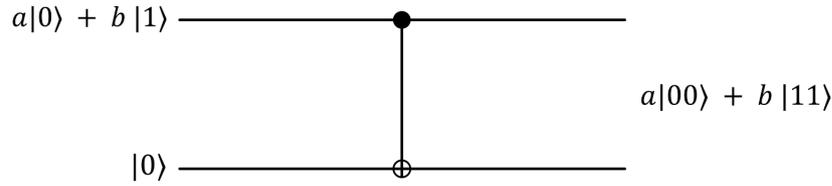
Fonte: Produzida pelos autores.

Observe que neste caso, o bit x , que pode assumir valores 0 ou 1, realmente foi clonado, pois quando aplica-se a porta XOR em um bit 0 e uma variável qualquer, pela tabela-verdade, a saída assumirá sempre o valor arbitrário da variável. Analogamente para os sistemas quânticos, temos a porta lógica CNOT. E se tentarmos clonar um estado quântico da mesma forma que clonamos um bit?

³ $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$

⁴ $\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$

Figura 7 – Clonagem de um estado quântico a partir da CNOT.



Fonte: Produzida pelos autores.

A porta CNOT atua em dois bits simultaneamente, usando primeiro bit como controle. Se seu valor for 0 ela não faz nada, se for 1, ela altera o valor do segundo bit:

$$|00\rangle \rightarrow |00\rangle, \quad (2.28)$$

$$|01\rangle \rightarrow |01\rangle, \quad (2.29)$$

$$|10\rangle \rightarrow |11\rangle, \quad (2.30)$$

$$|11\rangle \rightarrow |10\rangle. \quad (2.31)$$

Dessa maneira, observe que a informação que a e b fornece, é a amplitude de probabilidade de que cada estado ocorra, e após a medição, todas as informações que a antecedem são perdidas, inclusive a e b iniciais.

Outra maneira para demonstrar o Teorema da Não-Clonagem é lembrando do princípio da linearidade da superposição: se dois estados são solução de um sistema, então a combinação linear desses dois estados também é uma solução. Suponha que desejemos clonar um estado hipotético $|\phi\rangle = \alpha_0 |0\rangle + \beta_0 |1\rangle$. Para realizar a clonagem, que é uma cópia perfeita, usaremos um novo sistema quântico idêntico no mesmo espaço de Hilbert ⁵, com um estado inicial $|\psi\rangle = \alpha_1 |0\rangle + \beta_1 |1\rangle$. Este novo estado é independente do estado $|\phi\rangle$. O estado composto por $|\phi\rangle$ e $|\psi\rangle$ é o produto tensorial entre eles, que chamaremos de $|s\rangle$

$$|s\rangle = |\phi\rangle \otimes |\psi\rangle = |\phi\rangle |\psi\rangle. \quad (2.32)$$

Agora, devemos fazer uma observação: imagine que exista um operador capaz de copiar perfeitamente o estado de um qubit. O clone do estado $|\phi\rangle$ seria $|\phi\phi\rangle$:

$$|\phi\phi\rangle = (\alpha_0 |0\rangle + \beta_0 |1\rangle)(\alpha_0 |0\rangle + \beta_0 |1\rangle). \quad (2.33)$$

⁵ É um espaço vetorial dotado de produto interno, ou seja, com noções de distância e ângulos. Todo produto interno em um espaço vetorial real ou complexo, dá origem a uma norma: $\|x\| = \sqrt{\langle x, x \rangle}$. Chamamos H um espaço de Hilbert se estiver completo com relação a essa regra.

O clone do estado $|\psi\rangle$ seria $|\psi\psi\rangle$:

$$|\psi\psi\rangle = (\alpha_1 |0\rangle + \beta_1 |1\rangle)(\alpha_1 |0\rangle + \beta_1 |1\rangle). \quad (2.34)$$

Então, se esses dois estados puderam ser clonados, a saída para a superposição deles deveria ser a superposição das saídas $|O_{output}\rangle = \alpha |\phi\phi\rangle + \beta |\psi\psi\rangle$. Mas o que se deseja na verdade, é:

$$|s\rangle |s\rangle = (\alpha |\phi\rangle + \beta |\psi\rangle)(\alpha |\phi\rangle + \beta |\psi\rangle). \quad (2.35)$$

Essa é um cópia perfeita de $|s\rangle$. Desta maneira é observável que a impossibilidade de clonagem existe pela exigência de linearidade da superposição, pois o resultado que obteríamos, se houvesse um operador capaz de clonar perfeitamente um estado combinado, seria duas vezes este estado, que não corresponde a combinação linear dos estados iniciais.

Uma consequência deste teorema que a princípio pode parecer dificultar a teoria computacional quântica, é a criptografia. Suponhamos que Alice queira enviar várias informações através de qubits para Bob, que atuaria como receptor nesta hipótese. É possível que haja algum outro receptor a espera de obter informações de terceiros. Mesmo que este receptor indesejado consiga, em uma hipótese remota, clonar um estado específico, ainda assim ele não consegue ter acesso às informações enviadas pois em um estado superposto de qubits, a cópia perfeita é impossibilitada. Assim, a criptografia quântica gerada através da impossibilidade de clonagem de informação é um fator muito pertinente visto que na computação clássica, por mais que exista a criptografia, em alguns casos ela ainda pode ser quebrada. Quando falamos de informação quântica, é impossível que o mesmo aconteça.

2.1.5 Paralelismo quântico

Ao ouvir falar que um computador quântico da Google pôde resolver um problema de 10 mil anos em apenas 3 minutos, o questionamento de como a máquina quântica foi capaz de resolvê-lo tão rápido é iminente.

A computação clássica já resolve a maioria dos problemas que podem ser solucionados por máquinas, principalmente os supercomputadores atuais com processadores multinúcleo que tornam possível o processamento paralelo de dados. Neste tipo de processador, o sistema operacional trata cada núcleo como um processador diferente e isso permite que os vários núcleos trabalhem simultaneamente resolvendo cada um, uma instrução diferente. Ora, pensando por esse lado, pelo princípio de superposição, um sistema quântico pode estar simultaneamente em mais de um estado, ou seja, um computador quântico possui um grau de paralelismo natural, já que um qubit pode estar em um estado de superposição coerente entre os dois estados tradicionais de um bit (0 e 1). Uma boa

maneira para exemplificar e quantificar o paralelismo quântico, é por meio do algoritmo de *Deutsch* (DEUTSCH; JOZSA, 1992).

Seja a função $f(x)$, em que x possa assumir os valores binários 0 ou 1, definimos as funções $f(0)$ e $f(1)$ tais que:

$$\text{Solução} \Rightarrow \begin{cases} f(0) = 0 & \text{e } f(1) = 0, \\ f(0) = 1 & \text{e } f(1) = 1. \end{cases} \quad (2.36)$$

Desta forma, podemos classificar essa função como sendo constante quando $f(0) = f(1)$ e balanceada quando $f(0) \neq f(1)$. Com a computação clássica, resolveríamos esse problema comparando x e $f(x)$ sucessivamente. Esse algoritmo seria da seguinte forma:

Se $x = 0$ e $f(x) = 0$

Se $x_1 = 1$ e $f(x_1) = 0$

então a função é constante

senão

então a função é balanceada

Se $x = 0$ e $f(x) = 1$

Se $x_1 = 1$ e $f(x_1) = 1$

então a função é constante

senão

então a função é balanceada.

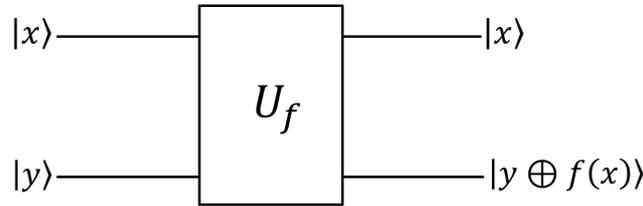
O computador fará comparações até que o problema seja solucionado. Note que ele precisa verificar sempre, pelo menos uma vez, os valores de x e $f(x)$. Isso tem um custo computacional, tanto de processamento, quanto de memória. Na computação clássica, quando analisamos um algoritmo para conhecer seu gasto operacional, levamos em consideração as principais operações que serão realizadas por ele. Em um algoritmo como esse, a comparação feita entre a função e o seu resultado é a principal operação executada. Deste modo, o computador realiza duas operações para cada possível valor de x .

Para apenas a verificação de um valor de x , o número de operações parece irrisório. Porém, imagine que precisaremos verificar várias funções, em que x pode assumir tanto 0 quanto 1, e $f(x)$ também. Duas operações para cada valor, já não parece mais tão insignificante.

O algoritmo de *Deutsch* propõe uma alternativa na computação quântica para a solução muito mais eficaz deste problema. Para demonstrá-lo, o primeiro passo é construir a operação unitária U_f que transforma o estado $|x\rangle|y\rangle$ em $|x\rangle|y \oplus f(x)\rangle$, em que \oplus é

conhecido como “ou exclusivo” ou adição módulo 2⁶.

Figura 8 – Operador U_f .



Fonte: Produzida pelos autores.

Para melhor visualização, se aplicarmos U_f em uma entrada padrão, ou seja, os estados da base $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$ obtemos:

$$U_f |00\rangle = |0\rangle |0 \oplus f(0)\rangle, \quad (2.37)$$

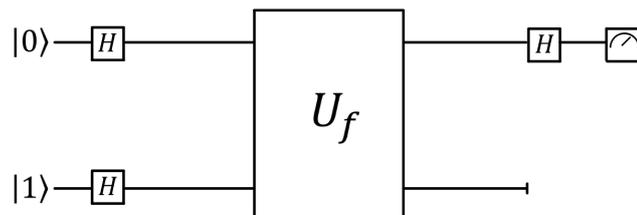
$$U_f |01\rangle = |0\rangle |1 \oplus f(0)\rangle, \quad (2.38)$$

$$U_f |10\rangle = |1\rangle |0 \oplus f(1)\rangle, \quad (2.39)$$

$$U_f |11\rangle = |1\rangle |1 \oplus f(1)\rangle. \quad (2.40)$$

O diferencial da computação quântica, é que ela nos permitirá, em vez de simplesmente iniciar com uma entrada padrão, usar portas lógicas quânticas para preparar uma entrada superposta de $|0\rangle$ e $|1\rangle$. Para preparar os qubits de entrada em uma superposição, usamos uma porta Hadamard atuando em cada um deles. Desse modo o operador U_f atua sobre o estado superposto, como mostra a figura 9, que apresenta o circuito responsável pela execução do algoritmo de *Deutsch*.

Figura 9 – Circuito de Deutsch.



Fonte: Produzida pelos autores.

⁶ Uma forma de aritmética utilizada na computação onde não existe a propagação de bits, ou seja, a soma de 1 com 1 resulta em 0, e não em 10, pois não há *carry* (sobe 1) entre as casas binárias na operação.

Uma porta Hadamard prepara o primeiro qubit $|0\rangle$ na superposição $(|0\rangle + |1\rangle)/\sqrt{2}$ e a outra prepara o segundo qubit $|1\rangle$ em $(|0\rangle - |1\rangle)/\sqrt{2}$. Para facilitar a visualização e a execução manual do algoritmo, primeiramente aplicaremos a operação U_f no estado generalizado $|x\rangle (|0\rangle - |1\rangle)/\sqrt{2}$. Desta forma, testaremos os possíveis valores de 0 e 1 a fim de encontrar uma equação geral. Temos que:

- Se $f(0) = 0$ e $f(1) = 0$

$$\text{Para } x = 0 \longrightarrow \frac{|0\rangle (|0\rangle - |1\rangle)}{\sqrt{2}} \quad (2.41)$$

$$U_f \frac{|0\rangle (|0\rangle - |1\rangle)}{\sqrt{2}} \Rightarrow \frac{(|0\rangle |0 \oplus f(0)\rangle - |0\rangle |1 \oplus f(0)\rangle)}{\sqrt{2}}. \quad (2.42)$$

Como neste caso, $f(0) = 0$, então $0 \oplus f(0) = 0$ e $1 \oplus f(0) = 1$. Prosseguindo, obtemos:

$$U_f \frac{|0\rangle (|0\rangle - |1\rangle)}{\sqrt{2}} = \frac{(|00\rangle - |01\rangle)}{\sqrt{2}}. \quad (2.43)$$

$$\text{Para } x = 1 \longrightarrow \frac{|1\rangle (|0\rangle - |1\rangle)}{\sqrt{2}} \quad (2.44)$$

$$U_f \frac{|1\rangle (|0\rangle - |1\rangle)}{\sqrt{2}} \Rightarrow \frac{(|1\rangle |0 \oplus f(1)\rangle - |1\rangle |1 \oplus f(1)\rangle)}{\sqrt{2}} \quad (2.45)$$

$$U_f \frac{|1\rangle (|0\rangle - |1\rangle)}{\sqrt{2}} = \frac{(|10\rangle - |11\rangle)}{\sqrt{2}}. \quad (2.46)$$

- Se $f(0) = 1$ e $f(1) = 1$

$$\text{Para } x = 0 \longrightarrow \frac{|0\rangle (|0\rangle - |1\rangle)}{\sqrt{2}} \quad (2.47)$$

$$U_f \frac{|0\rangle (|0\rangle - |1\rangle)}{\sqrt{2}} \Rightarrow \frac{(|0\rangle |0 \oplus f(0)\rangle - |0\rangle |1 \oplus f(0)\rangle)}{\sqrt{2}}. \quad (2.48)$$

Neste caso, $f(0) = 1$, então $0 \oplus f(0) = 1$ e $1 \oplus f(0) = 0$. Prosseguindo, obtemos:

$$U_f \frac{|0\rangle (|0\rangle - |1\rangle)}{\sqrt{2}} = \frac{(|01\rangle - |00\rangle)}{\sqrt{2}}. \quad (2.49)$$

$$\text{Para } x = 1 \longrightarrow \frac{|1\rangle (|0\rangle - |1\rangle)}{\sqrt{2}} \quad (2.50)$$

$$U_f \frac{|1\rangle (|0\rangle - |1\rangle)}{\sqrt{2}} \Rightarrow \frac{(|1\rangle |0 \oplus f(1)\rangle - |1\rangle |1 \oplus f(1)\rangle)}{\sqrt{2}} \quad (2.51)$$

$$U_f \frac{|1\rangle (|0\rangle - |1\rangle)}{\sqrt{2}} = \frac{(|11\rangle - |10\rangle)}{\sqrt{2}}. \quad (2.52)$$

- Se $f(0) = 0$ e $f(1) = 1$

$$\text{Para } x = 0 \longrightarrow \frac{|0\rangle (|0\rangle - |1\rangle)}{\sqrt{2}} \quad (2.53)$$

$$U_f \frac{|0\rangle (|0\rangle - |1\rangle)}{\sqrt{2}} \Rightarrow \frac{(|0\rangle |0 \oplus f(0)\rangle - |0\rangle |1 \oplus f(0)\rangle)}{\sqrt{2}}. \quad (2.54)$$

Agora, temos que $f(0) = 0$, então, $0 \oplus f(0) = 0$ e $1 \oplus f(0) = 1$. Prosseguindo, obtemos:

$$U_f \frac{|0\rangle (|0\rangle - |1\rangle)}{\sqrt{2}} = \frac{(|01\rangle - |00\rangle)}{\sqrt{2}}. \quad (2.55)$$

$$\text{Para } x = 1 \longrightarrow \frac{|1\rangle (|0\rangle - |1\rangle)}{\sqrt{2}} \quad (2.56)$$

$$U_f \frac{|1\rangle (|0\rangle - |1\rangle)}{\sqrt{2}} \Rightarrow \frac{(|1\rangle |0 \oplus f(1)\rangle - |1\rangle |1 \oplus f(1)\rangle)}{\sqrt{2}}. \quad (2.57)$$

Temos que $f(1) = 1$, então, $0 \oplus f(1) = 1$ e $1 \oplus f(1) = 0$. Prosseguindo, obtemos:

$$U_f \frac{|1\rangle (|0\rangle - |1\rangle)}{\sqrt{2}} = \frac{(|11\rangle - |10\rangle)}{\sqrt{2}}. \quad (2.58)$$

- Se $f(0) = 1$ e $f(1) = 0$

$$\text{Para } x = 0 \longrightarrow \frac{|0\rangle (|0\rangle - |1\rangle)}{\sqrt{2}} \quad (2.59)$$

$$U_f \frac{|0\rangle (|0\rangle - |1\rangle)}{\sqrt{2}} \Rightarrow \frac{(|0\rangle |0 \oplus f(0)\rangle - |0\rangle |1 \oplus f(0)\rangle)}{\sqrt{2}}. \quad (2.60)$$

Temos que $f(0) = 1$, então $0 \oplus f(0) = 1$ e $1 \oplus f(0) = 0$. Prosseguindo, obtemos:

$$U_f \frac{|0\rangle (|0\rangle - |1\rangle)}{\sqrt{2}} = \frac{(|01\rangle - |00\rangle)}{\sqrt{2}}. \quad (2.61)$$

$$\text{Para } x = 1 \longrightarrow \frac{|1\rangle (|0\rangle - |1\rangle)}{\sqrt{2}} \quad (2.62)$$

$$U_f \frac{|1\rangle(|0\rangle - |1\rangle)}{\sqrt{2}} \Rightarrow \frac{(|1\rangle|0 \oplus f(1)\rangle - |1\rangle|1 \oplus f(1)\rangle)}{\sqrt{2}}. \quad (2.63)$$

Por fim, temos que $f(1) = 0$, desta forma, $0 \oplus f(1) = 0$ e $1 \oplus f(1) = 1$. Prosseguindo, obtemos:

$$U_f \frac{|1\rangle(|0\rangle - |1\rangle)}{\sqrt{2}} = \frac{(|10\rangle - |11\rangle)}{\sqrt{2}}. \quad (2.64)$$

A intenção ao realizar esses cálculos de forma manual, é encontrar uma equação geral para demonstrar como o computador quântico pode ser capaz calcular a função em apenas uma execução. Assim, devemos observar o que existe em comum entre todos os resultados encontrados ao aplicar o operador U_f nas entradas possíveis. Analisando, chegamos à conclusão que qualquer entrada pode ser descrita seguindo a equação:

$$(-1)^{f(x)} |x\rangle \frac{(|0\rangle - |1\rangle)}{\sqrt{2}}. \quad (2.65)$$

Com a equação generalizada, vamos verificar agora, a entrada superposta obtida através portas Hadamard na entrada do circuito de *Deutsch*, como mostra a figura 9. Como o estado inicial de $|x\rangle|y\rangle$, é $|0\rangle|1\rangle$, a entrada que o operador U_f recebe é um estado que pode ser descrito como:

$$|I_{input}\rangle = \left[\frac{|0\rangle + |1\rangle}{\sqrt{2}} \right] \otimes \left[\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right]. \quad (2.66)$$

Calculando o produto tensorial, temos:

$$|I_{input}\rangle = \frac{|00\rangle}{2} - \frac{|01\rangle}{2} + \frac{|10\rangle}{2} - \frac{|11\rangle}{2} \quad (2.67)$$

$$|I_{input}\rangle = \frac{|0\rangle}{2}(|0\rangle - |1\rangle) + \frac{|1\rangle}{2}(|0\rangle - |1\rangle). \quad (2.68)$$

Assim, chegamos em um estado de superposição linear de $|0\rangle$ e $|1\rangle$, em que cada porção da equação é equivalente à probabilidade de o valor da entrada x ser 0 ou 1. Com isso, o próximo passo é substituir os valores na equação 2.65, já que ela representa uma generalização para a aplicação do operador U_f

$$U_f |I_{input}\rangle = (-1)^{f(0)} \frac{|0\rangle}{2}(|0\rangle - |1\rangle) + (-1)^{f(1)} \frac{|1\rangle}{2}(|0\rangle - |1\rangle). \quad (2.69)$$

Se a função é constante, ou seja, $f(0) = f(1)$, os fatores $(-1)^{f(0)}$ e $(-1)^{f(1)}$ tornam-se um fator comum, já que os expoentes são iguais, sejam eles 0 ou 1. Sendo um múltiplo da soma, podemos colocá-lo em evidência e já que são iguais, pode ser tanto $f(0)$ quanto $f(1)$, mudando apenas a escrita

$$U_f |I_{input}\rangle = (-1)^{f(0)} \left[\frac{|0\rangle}{2}(|0\rangle - |1\rangle) + \frac{|1\rangle}{2}(|0\rangle - |1\rangle) \right]. \quad (2.70)$$

A porção da equação que está dentro dos colchetes, é exatamente a entrada superposta gerada através das portas Hadamard no começo do circuito, como vemos nas equações 2.66, 2.67 e 2.68. Assim, podemos reescrevê-la, já que o primeiro qubit ainda deve passar por uma porta Hadamard ao final do circuito, para que então, possa ser medido

$$U_f |I_{input}\rangle = (-1)^{f(0)} \left[\frac{|0\rangle + |1\rangle}{\sqrt{2}} \right] \left[\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right]. \quad (2.71)$$

Rearranjado, o estado gerado pelo operador U_f está pronto para passar pela porta Hadamard. Ora, se observarmos na figura 9, apenas o primeiro qubit passará pela porta, ou seja, apenas $(|0\rangle + |1\rangle)/\sqrt{2}$. Este é o estado $|0\rangle$ quando aplicada Hadamard. Dessa forma, o que acontece quando aplicamos novamente a porta é justamente o processo inverso, e o qubit volta ao seu valor $|0\rangle$.

Se a função é balanceada, ou seja $f(0) \neq f(1)$, os fatores $(-1)^{f(0)}$ e $(-1)^{f(1)}$, são diferentes, onde $f(0) = 0$ e $f(1) = 1$ ou $f(0) = 1$ e $f(1) = 0$. Desta forma, para saber o que acontece com a equação quando aplicada a porta Hadamard, calculamos:

- Se $f(0) = 0$ e $f(1) = 1$

$$U_f |I_{input}\rangle = (-1)^0 \frac{|0\rangle}{2}(|0\rangle - |1\rangle) + (-1)^1 \frac{|1\rangle}{2}(|0\rangle - |1\rangle) \quad (2.72)$$

$$U_f |I_{input}\rangle = \frac{|0\rangle}{2}(|0\rangle - |1\rangle) + (-1) \frac{|1\rangle}{2}(|0\rangle - |1\rangle) \quad (2.73)$$

$$U_f |I_{input}\rangle = \frac{|00\rangle}{2} - \frac{|01\rangle}{2} - \frac{|10\rangle}{2} + \frac{|11\rangle}{2} \quad (2.74)$$

$$U_f |I_{input}\rangle = \left[\frac{(|0\rangle - |1\rangle)}{\sqrt{2}} \right] \left[\frac{(|0\rangle - |1\rangle)}{\sqrt{2}} \right]. \quad (2.75)$$

Observe que agora, a porta Hadamard será aplicada ao primeiro qubit $(|0\rangle - |1\rangle)/2$, e esse estado é a superposição criada ao aplicar-se essa porta ao qubit de estado inicial $|1\rangle$, assim, a operação inversa será realizada e o qubit retornará ao valor $|1\rangle$.

- Se $f(0) = 1$ e $f(1) = 0$

$$U_f |I_{input}\rangle = (-1)^1 \frac{|0\rangle}{2}(|0\rangle - |1\rangle) + (-1)^0 \frac{|1\rangle}{2}(|0\rangle - |1\rangle) \quad (2.76)$$

$$U_f |I_{input}\rangle = (-1) \frac{|0\rangle}{2} (|0\rangle - |1\rangle) + \frac{|1\rangle}{2} (|0\rangle - |1\rangle) \quad (2.77)$$

$$U_f |I_{input}\rangle = -\frac{|00\rangle}{2} + \frac{|01\rangle}{2} + \frac{|10\rangle}{2} - \frac{|11\rangle}{2} \quad (2.78)$$

$$U_f |I_{input}\rangle = (-1) \left[\frac{(|0\rangle - |1\rangle)}{\sqrt{2}} \right] \left[\frac{(|0\rangle - |1\rangle)}{\sqrt{2}} \right]. \quad (2.79)$$

O qubit a ser medido após a aplicação da Hadamard é $|1\rangle$, adicionando um coeficiente (-1) , pois obtemos como estado após $U_f |I_{input}\rangle$, $(|0\rangle - |1\rangle)/2$.

Feito isso, a conclusão em que chegamos é que quando a função é constante, ao final do algoritmo medimos $|0\rangle$, e quando a função é balanceada, medimos $|1\rangle$. Observe que $f(0) \oplus f(1) = 0$ se $f(0) = f(1)$ e $f(0) \oplus f(1) = 1$ se $f(0) \neq f(1)$. Com apenas uma medida, o computador quântico responde se a função é constante ou balanceada enquanto o computador clássico necessitaria de duas medidas para cada x . Isto é, caímos de um problema com resolução polinomial, para um problema com resolução linear e quando tratamos de computação, isso é excelente, e este é o motivo de a computação quântica ser tão eficaz para certos problemas.

A possibilidade de a unidade básica do computador quântico encontrar-se em uma superposição como propriedade natural e intrínseca, permite com que algumas soluções sejam exponencialmente mais rápidas que em um computador clássico. Com essa vantagem exponencial de processamento paralelo, a computação quântica traz grandes promessas no que diz respeito à Inteligência Artificial e ao aprendizado de máquina.

2.2 Redes Neurais Artificiais

Componentes essenciais da Inteligência Artificial, as Redes Neurais Artificiais são algoritmos que apresentam um conjunto de técnicas com a intenção de mapear o cérebro humano em modelos matemáticos, para que esses modelos sejam capazes de processar informação, tomar decisões e aprender de alguma forma. Inspiram-se nos neurônios biológicos e sistema nervoso, de forma análoga à rede neural biológica, podemos dizer que, uma rede neural artificial é formada por vários neurônios artificiais que estão ligados por conexões, conhecidas como conexões sinápticas. Desta maneira, entendendo um neurônio artificial, entende-se o cerne de uma rede neural, e é isto que faremos nesta seção ao estudar o Perceptron.

2.2.1 Perceptron Clássico

Um dos modelos mais clássicos e simples de neurônio artificial, o Perceptron foi desenvolvido na década de 50 pelo psicólogo norte-americano Frank Rosenblatt, que para isso baseou-se em trabalhos anteriores de McCulloch e Pitts ([MCCLULLOCH; PITTS,](#)

1943). Foi pensado para lidar com o problema de reconhecimento de padrões, tarefa que é executada facilmente por nós, humanos, mas que é um grande desafio para as máquinas.

Podemos descrevê-lo como um modelo matemático classificador, no qual as entradas recebidas pelo algoritmo são estudadas e padronizadas de forma a serem classificadas como pertencentes à uma entre duas possíveis classes de saída. O Perceptron simples que estamos descrevendo aqui, é um modelo bom para os dados que podem ser separados linearmente, de modo que seja possível classificá-los nos valores binários 0 ou 1. Para o funcionamento e aprendizado do algoritmo são usados dados supervisionados, ou seja, dados que possuem uma saída predeterminada para cada possível padrão de entrada. Esses dados são compostos por N registros, cada um deles possuindo um vetor de tamanho j que servirá como entrada para o algoritmo. Esse vetor é composto por valores binários que são respostas à diferentes estímulos (em computação é padrão representar a resposta verdadeira como 1 e a falsa como 0), e por fim, cada registro possui uma saída respectiva, que é gerada como resposta às combinações possíveis de valores do vetor i . Recapitulando: os dados podem conter N registros, cada registro contém um vetor com j posições que, quando analisadas, geram uma saída.

O método com que o algoritmo procura uma reta afim de separar os dados é utilizando o vetor de entrada e um vetor de pesos, cujas posições correspondem à um valor do vetor de entrada e somando um termo independente b . Esse termo independente é conhecido como *bias*, e é responsável pelo deslocamento da reta por um fator, o que pode ser fundamental para a classificação de certos conjuntos de dados. Assim, para cada registro, para um vetor de entrada de tamanho j , haverá um vetor de pesos correspondente de tamanho j . Sistemáticamente, podemos descrever os passos da implementação do algoritmo. É realizada soma entre o *bias* o somatório da multiplicação escalar dos valores de entrada e os valores de peso

$$x = b + \sum_{j=0}^{N-1} w_j \cdot i_j \quad (2.80)$$

em que j corresponde ao índice da posição no vetor.

Tendo realizado este somatório, precisamos de uma função que é chamada “função de ativação”. É ela que determina como o dado é classificado *a priori*. A função de ativação que utilizaremos é a função degrau

$$\hat{y} = f(x) \Rightarrow \begin{cases} \hat{y} = 0 & \text{se } x \leq 0, \\ \hat{y} = 1 & \text{se } x > 0. \end{cases} \quad (2.81)$$

Deste modo, basta verificar se a saída fornecida pela função de ativação corresponde à saída desejada. Se são correspondentes, o algoritmo segue seu curso, verificando as próximas entradas. Se as saídas forem diferentes, necessitamos que o algoritmo calcule o erro e aprenda o padrão correto, ajustando para este fim, o vetor de pesos. O erro é calculado

subtraindo o valor esperado da resposta pelo valor encontrado pela função de ativação, ou seja, $y - \hat{y}$, e o peso ajustado será

$$w = w_j + \eta \cdot (y - \hat{y}) \cdot i_j \quad (2.82)$$

em que η é a taxa de aprendizado, definida pelo programador. Faz-se isso, para todos os valores do vetor peso. Depois de todos os w_j ajustados, o algoritmo continuará no próximo valor de entrada. Se necessário ajusta-se novamente o vetor de pesos, mas sempre continuando pelo próximo valor no vetor i . Quando toda a entrada tiver sido percorrida, repete-se a operação de verificação de saídas, até que os valores de vetor peso sejam válidos para todos os N dados de entrada e o Perceptron tenha aprendido o padrão e responda corretamente para todos o vetor i . Uma limitação apresentada por este modelo de neurônio é que ele é um bom modelo, mas apenas para os dados que são linearmente separáveis.

2.3 Trabalhos Relacionados

[Lewenstein \(1994\)](#), ainda na década de 90, definiu os passos para a construção de modelos quânticos para o Perceptron como sendo: *(i)* a preparação dos estados de entrada; *(ii)* a transformação unitária de estados de entrada em estados de saída e *(iii)* a medição da saída. Esse trabalho foi um dos pioneiros sobre o tema e antes mesmo de possuímos hardwares de processadores quânticos construídos, já buscava responder se existe a possibilidade de um modelo de Perceptron quântico responder corretamente à determinados padrões de entrada.

Posteriormente, [Zhou, Qin e Jiang \(2006\)](#) apresentaram resultados e análise estatística de que modelos quânticos do Perceptron, são capazes de classificar corretamente até mesmo padrões não necessariamente lineares, utilizando a porta lógica XOR, um exemplo vastamente utilizado na literatura da computação para a demonstração da limitação do Perceptron clássico.

Mais recentemente, utilizando os mesmos passos indicados na teoria por [Lewenstein \(1994\)](#), [Tacchino et al. \(2019\)](#) propuseram o modelo utilizado como base para este trabalho realizando as simulações em um processador quântico de 5 qubits, obtendo resultados que confirmam que o Perceptron quântico, concorda com o modelo estatístico e responde corretamente aos padrões de entrada. Em sua tese de mestrado, [Bon \(2019\)](#) teve como base esse mesmo modelo e realizou simulações de classificação das letras do alfabeto, outro experimento que confirma a capacidade da versão quântica romper a exigência de dados linearmente separáveis, existente no modelo clássico do Perceptron.

3 Desenvolvimento

3.1 Perceptron Quântico

Por sua simplicidade de execução e implementação, o Perceptron é um algoritmo de Redes Neurais Artificiais ideal a se criar uma versão quântica. Por outro lado, por mais simples que seja sua implementação, à medida que aumentam o número de nós e as conexões entre camadas, o custo computacional aumenta também. Isso faz com que se busque maneiras para otimizar o funcionamento do algoritmo. Dessa forma, várias estratégias podem ser utilizadas e, juntando o útil ao agradável, a computação quântica é uma estratégia teoricamente eficaz e várias propostas já foram apresentadas nos últimos anos para a sua implementação em computadores quânticos. (WIEBE; KAPOOR; SVORE, 2016)

Utilizando cada qubit como neurônio individual dentro da rede, podemos explorar a vantagem exponencial do armazenamento da informação quântica. Como vimos na seção 2.1.2, usando N qubits, possuiremos 2^N valores de entrada. Dessa forma já se obtém um ganho se comparado à versão clássica, pois para os vetores de entrada e peso, em vez de N valores, são calculados 2^N valores.

Para a implementação do Perceptron quântico, nos basearemos no artigo “An artificial neuron implemented on an actual quantum processor” (TACCHINO et al., 2019), em que o estudo estará restringido ao caso em que os vetores de entrada (\vec{i}) e peso (\vec{w}) assumem os valores $i_j, w_j \in \{-1, 1\}$. A partir daqui nos referiremos a $m = 2^N$ e definiremos as funções de entrada e peso como m -dimensionais. Dados os vetores arbitrários de entrada e peso

$$\vec{i} = \begin{bmatrix} i_0 \\ i_1 \\ \vdots \\ i_{m-1} \end{bmatrix}, \quad \vec{w} = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_{m-1} \end{bmatrix} \quad (3.1)$$

podemos definir os estados de entrada e peso como

$$\begin{aligned} |\psi_i\rangle &= \frac{1}{\sqrt{m}} \sum_{j=0}^{m-1} i_j |j\rangle \\ |\psi_w\rangle &= \frac{1}{\sqrt{m}} \sum_{j=0}^{m-1} w_j |j\rangle \end{aligned} \quad (3.2)$$

em que $1/\sqrt{m}$ é a amplitude de probabilidade. Vale observar que o estado $|j\rangle \in \{|00\dots 00\rangle, |00\dots 01\rangle, \dots, |11\dots 11\rangle\}$, ou seja, todas as possibilidades de estados que um qubit pode

assumir estando em $|0\rangle$ ou $|1\rangle$ com m combinações diferentes. Na base decimal, essa representação seria equivalente a $j \in \{0, 1, \dots, m - 1\}$.

Tendo estabelecido a forma de $|\psi_i\rangle$, o primeiro passo é prepará-lo. Assumindo que os qubits serão inicializados no estado $|00 \dots 00\rangle \equiv |0\rangle^{\otimes N}$ precisamos de uma porta lógica que realize a transformação unitária da seguinte forma:

$$U_i |0\rangle^{\otimes N} = |\psi_i\rangle. \quad (3.3)$$

A princípio, qualquer matriz unitária m -dimensional quadrada, com o vetor \vec{i} na primeira coluna, é adequada para este propósito. Posteriormente, proporemos uma forma explícita para U_i . O próximo passo é calcular o produto interno entre \vec{i} e \vec{w} e para realizá-lo de forma eficiente, podemos definir também uma porta U_w que realize a transformação unitária do estado quântico de pesos de tal forma que:

$$U_w |\psi_w\rangle = |1\rangle^{\otimes N} = |m - 1\rangle. \quad (3.4)$$

A transformação realizada pelo operador U_w , pode não parecer muito intuitiva, mas a figura 10, ilustra geometricamente o que ocorre ao se aplicar o operador U_w nos estados quânticos de entrada e peso. Observe que U_w rotaciona $|\psi_w\rangle$ até que se alinhe com o eixo $|111 \dots 11\rangle$, rotacionando também $|\psi_i\rangle$ pelo mesmo ângulo. Dessa forma, o produto interno entre $|\psi_i\rangle$ e $|\psi_w\rangle$ seja a projeção de $U_w |\psi_i\rangle$ sobre o eixo $|111 \dots 11\rangle$. Essa manobra de rotação é comum mesmo na física clássica e na matemática, de modo a facilitar a realização do produto escalar. Isso porque em uma base qualquer não é tão trivial realizá-lo, ao contrário do que acontece sobre um eixo determinado e de fácil manejo.

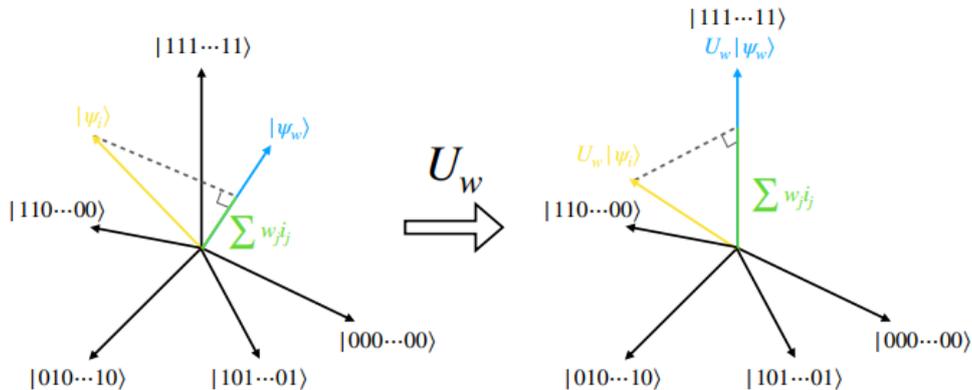


Figura 10 – Representação geométrica de U_w atuando sobre os estados $|\psi_i\rangle$ e $|\psi_w\rangle$.

Algebricamente, aplicando U_w em $|\psi_i\rangle$, obtemos

$$U_w |\psi_i\rangle = \sum_{j=0}^{m-1} c_j |j\rangle \equiv |\phi_{i,w}\rangle \quad (3.5)$$

em que $|\phi_{i,w}\rangle$ é o estado rotacionado ao se aplicar U_w e $\sum_{j=0}^{m-1} |c_j|^2 = 1$. Os coeficientes c_j dependem da forma particular de U_w .

O produto interno entre $U_w |\psi_i\rangle$ e $U_w |\psi_w\rangle$ é

$$\langle \psi_w | U_w^\dagger U_w |\psi_i\rangle = \langle \psi_w | I |\psi_i\rangle = \langle \psi_w | \psi_i\rangle = \langle m-1 | \phi_{i,w}\rangle = c_{m-1} \quad (3.6)$$

Isso só é possível porque U_w é uma matriz unitária; quando fazemos o produto interno entre $U_w |\psi_i\rangle$ e $U_w |\psi_w\rangle$ equivale a calcular $\langle \psi_i | U_w^\dagger U_w |\psi_w\rangle$, em que U_w^\dagger é o conjugado Hermitiano¹ de U_w e $U_w^\dagger U_w = I$, para que U_w seja unitário. Lembrando que ter $c_{m-1} = 1$ corresponde a $|\psi_i\rangle = |\psi_w\rangle$, o que possuímos como produto interno é o mesmo que $\| |\psi_{i,j}\rangle \| \cos 0 = 1$.

Calculando agora o mesmo produto a partir das definições em (3.2), temos

$$\langle \psi_i | \psi_w\rangle = \frac{1}{m} \sum_{k,j} i_k w_j \langle j | k\rangle = \frac{1}{m} \sum_{k,j} i_k w_j \delta_{k,j} = \frac{1}{m} \vec{i} \cdot \vec{w} \quad (3.7)$$

Esse resultado mostra que o produto escalar entre os vetores peso e saída é $\vec{w} \cdot \vec{i} = m \langle \psi_w | \psi_i\rangle$. Portanto, o resultado desejado está contido, até um fator de normalização, no coeficiente c_{m-1} do estado final $|\phi_{i,w}\rangle$ - ver figura 10.

Para extrair alguma informação, a proposta é utilizar o qubit *ancilla* (um qubit auxiliar) inicializado no estado $|0\rangle$. Utilizando uma porta CNOT atuando nos N qubits que trabalham como qubits controlados, enquanto o *ancilla* trabalha como o alvo, obtem-se

$$\begin{aligned} \text{CNOT}(|\phi_{i,w}\rangle \otimes |0\rangle_a) &= \text{CNOT} \left(\sum_{j=0}^{m-1} c_j |j\rangle \otimes |0\rangle_a \right) \\ &= \sum_{j=0}^{m-2} c_j |j\rangle \otimes |0\rangle_a + c_{m-1} |m-1\rangle \otimes |1\rangle_a \end{aligned} \quad (3.8)$$

em que os itens com subscrição a representam o qubit ancilla. Observe que forçamos o sistema a fazer o qubit *ancilla*, na última camada do Perceptron, atuando como alvo, seja alterado de $|0\rangle$ para $|1\rangle$. Isso só ocorre se quando aplicada a porta CNOT, o qubit controlado seja $|1\rangle$. Desta forma, podemos estimar o valor de $|c_{m-1}|^2$, pois já sabemos que medimos $|1\rangle$, ou seja, “Perceptron ativado”, com probabilidade $|c_{m-1}|^2$.

3.1.1 Construção do modelo

Como proposto pelos autores (TACCHINO et al., 2019), é necessário construir um circuito quântico de forma a se obter os operadores U_i e U_w . Partindo do estado $|0\rangle^N$,

¹ A matriz que representa esse operador é a conjugada transposta, neste caso também pode ser escrita como $(\bar{U})^T$, onde \bar{U} representa o conjugado da parte complexa da matriz U , e U^T representa o transposto, que é a matriz que surge a partir de U quando trocamos as linhas pelas colunas e vice-versa.

deseja-se alcançar o estado superposto $|\psi_i\rangle$. Como visto ao longo deste trabalho, é possível obter um estado superposto através da utilização da porta lógica Hadamard, desta forma, é necessário que ao início do circuito, exista um bloco do tipo $H^{\otimes N}$. Feita a superposição, é proposto que, para que se possa reproduzir um vetor arbitrário de entrada \vec{i} seguindo as especificações dadas, ou seja, $i_j, w_j \in \{-1, 1\}$, sejam utilizadas portas Z controladas² para inverter o sinal de determinados qubits. O bloco composto pela porta Z controlada, ou Signal Flip (SF), deve atuar da seguinte maneira:

$$SF_{N,j} |j'\rangle \begin{cases} |j'\rangle & \text{se } j \neq j', \\ -|j'\rangle & \text{se } j = j'. \end{cases} \quad (3.9)$$

A configuração deste bloco composto pelas portas CZ varia de acordo com a quantidade de qubits utilizada e do vetor \vec{i} pretendido. Por exemplo, em um circuito para 2 qubits em que se deseja reproduzir $\vec{i} = (1, 1, -1, 1)$, o bloco se reduz a apenas uma porta CZ . No caso do exemplo com $N = 2 \implies m = 4$, o estado $|\psi_i\rangle$ será dado por:

$$|\psi_i\rangle = \frac{1}{\sqrt{4}} \sum_{j=0}^3 i_j |j\rangle. \quad (3.10)$$

Em que $|j\rangle$ assumirá os valores $|00\rangle, |01\rangle, |10\rangle, |11\rangle$. Olhando para o vetor \vec{i} almejado, o terceiro coeficiente neste caso, deve assumir valor negativo. Neste ponto, é que o bloco SF é importante: apenas o terceiro qubit deve ter seu coeficiente invertido. Para este fim, algumas soluções podem ser aplicadas (TACCHINO et al., 2019), mas, utilizando uma porta $CNOT$ seguida de uma porta CZ e reaplicando a $CNOT$ é possível obter exatamente a configuração desejada.

$$|00\rangle, |01\rangle, |10\rangle, |11\rangle \xrightarrow{CNOT} |00\rangle, |01\rangle, |11\rangle, |10\rangle \quad (3.11)$$

$$|00\rangle, |01\rangle, |11\rangle, |10\rangle \xrightarrow{CZ} |00\rangle, |01\rangle, -|11\rangle, |10\rangle \quad (3.12)$$

$$|00\rangle, |01\rangle, -|11\rangle, |10\rangle \xrightarrow{CNOT} |00\rangle, |01\rangle, -|10\rangle, |11\rangle \quad (3.13)$$

Seguindo esse raciocínio, é possível expandir para N qubits.

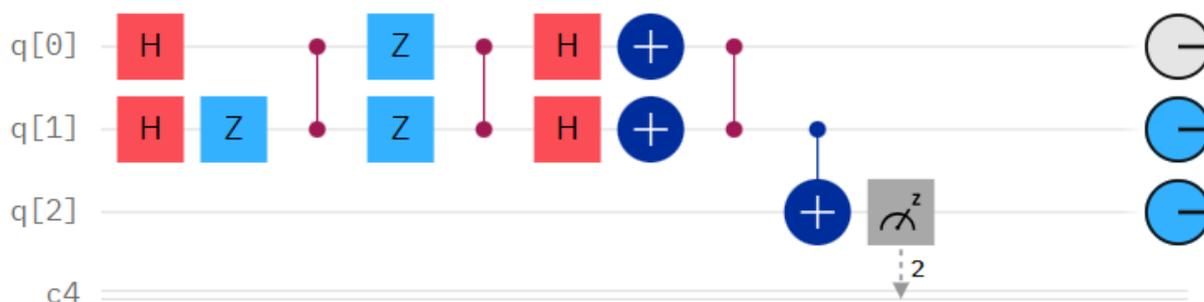
O passo seguinte, é encontrar a composição do operador U_w , de modo que $U_w |\psi_w\rangle = |1\rangle^{\otimes N}$. Ora, é intuitivo pensar que, encontrando $|\psi_w\rangle$ como um estado superposto com distribuição uniforme das probabilidades, em que todos os qubits estão no estado $|+\rangle$, aplicando a porta Hadamard, tem-se o estado $|0\rangle^{\otimes N}$ e com isso, aplicando $X^{\otimes N}$, o estado $|1\rangle^{\otimes N}$ é alcançado.

$$|\psi_w\rangle \longrightarrow |\psi_+\rangle \xrightarrow{H^{\otimes N}} |0\rangle^{\otimes N} \xrightarrow{X^{\otimes N}} |1\rangle^{\otimes N} \quad (3.14)$$

² Ou seja, portas Z que, assim como a porta CNOT, atuem utilizando um qubit de controle.

Tendo descrito os operadores U_i e U_w , é possível criar o diagrama de blocos desse modelo, para este fim, foi utilizado o simulador online oferecido pela IBM³. A Figura 11 mostra o diagrama obtido através das descrições feitas ao longo desta seção.

Figura 11 – Possível arranjo para um circuito quântico que representa o Perceptron para $N = 2$.



Fonte: Produzida pelos autores.

³ <https://quantum-computing.ibm.com/composer>

4 Resultados

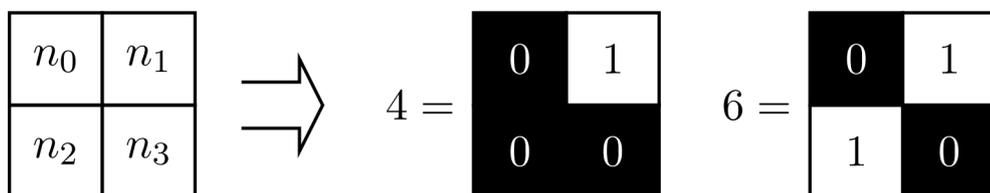
Neste capítulo serão explicitados e analisados os resultados obtidos através das simulações realizadas pelos autores do artigo “*An artificial neuron implemented on an actual quantum processor*” (TACCHINO et al., 2019) e também obtidos pelos autores da tese de mestrado “*Quantum algorithm for the implementation of a perceptron*” (BON, 2019). Ambos realizaram experimentos utilizando o algoritmo proposto neste trabalho, sendo que a tese, assim como esta monografia, teve como base o estudo realizado no artigo. Além disso, também será demonstrada e analisada a simulação numérica do experimento que propomos e realizamos, cujo objetivo é utilizar o Perceptron para identificar cores de 8 bits.

4.0.1 Evidências computacionais

Como na seção de desenvolvimento (3), a princípio será utilizado um circuito de 2 qubits. Dessa maneira, como visto, é possível representar vetores de tamanho 4 ($2^2 = 4$). Além disso, como proposto, \vec{i}, \vec{w} estão compreendidos nos valores 1 e -1, portanto, o experimento consiste em 2^4 combinações de \vec{i} e \vec{w} .

Para uma compreensão visual da representação dos vetores de entrada e peso, é possível representá-los em imagens de 4 *pixels* preenchidos pelas cores branca, representando o valor -1, e preta, representando o valor 1. Lembrando que, para o algoritmo proposto, 1 é o equivalente ao valor binário 0 e -1 é equivalente ao valor binário 1. Dessa maneira, os valores $4 = \{0, 1, 0, 0\} \equiv \{1, -1, 1, 1\}$ e $6 = \{0, 1, 1, 0\} \equiv \{1, -1, -1, 1\}$, podem ser representados como mostra a Figura 12.

Figura 12 – Exemplo de representação dos vetores de entrada e peso.

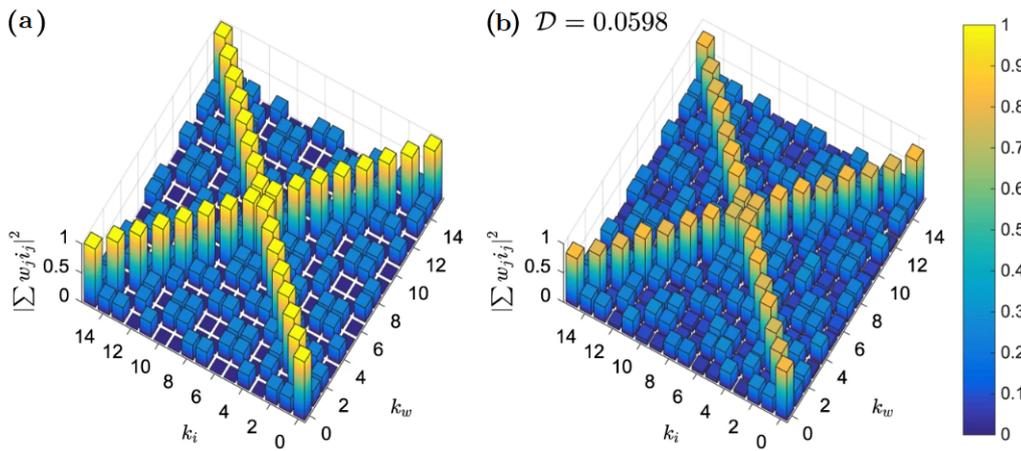


Fonte: (BON, 2019, p. 13)

Nesta representação, temos que um vetor arbitrário $\vec{n} = \{n_0, n_1, n_2, n_3\}$ é dado pelo preenchimento dos *pixels* na ordem do bit mais significativo para o menos significativo da esquerda para a direita e de cima para baixo. É possível expandir essa representação para qualquer número de qubits, ou seja, vetores de qualquer tamanho.

A princípio, foram realizados experimentos variando os possíveis valores de peso e entrada para $N = 2$, que resultam em um total de 16 valores possíveis para \vec{i} e \vec{w} . Para isso, foi utilizando o processador IBM Quantum 5 Tenerife¹ em conjunto com o Qiskit². A Figura 13 mostra o conjunto de histogramas que representam os resultados desta simulação.

Figura 13 – **a.** Resultado ideal para o Perceptron, simulado em um processador clássico. **b.** Resultado da simulação simulada em um processador quântico. Além disso, é explicitado o valor da discrepância média correspondente em relação ao caso ideal.



Fonte: (TACCHINO et al., 2019, p. 4)

O histograma apresentado na Figura 13 (a) mostra o resultado ideal para o algoritmo do Perceptron quântico. Observe que, tal resultado reflete exatamente o estudo realizado no capítulo 3. Para qualquer valor de \vec{w} , o Perceptron é capaz de classificar perfeitamente, ou seja, $|c_{m-1}|^2$, padrões de entrada em que $\vec{i} = \vec{w}$, enquanto todos os outros casos apresentam saída menor do que 0,25. Dito isso e pensando na forma sugerida na Figura 12, é plausível perceber que, um único Perceptron pode ser capaz de identificar padrões que podem ser representados por *pixels*, como será demonstrado em um resultado posterior. O histograma apresentado em 13 (b) mostra o resultado obtidos através das simulações no IBM Q 5 Tenerife. É notável que esse caso apresenta resultados precisos se comparado ao ideal, de maneira que todas as combinações de \vec{i} e \vec{w} apresentam resultados maiores que 0,75 ou menores que 0,3, produzindo classificações corretas para todos os casos.

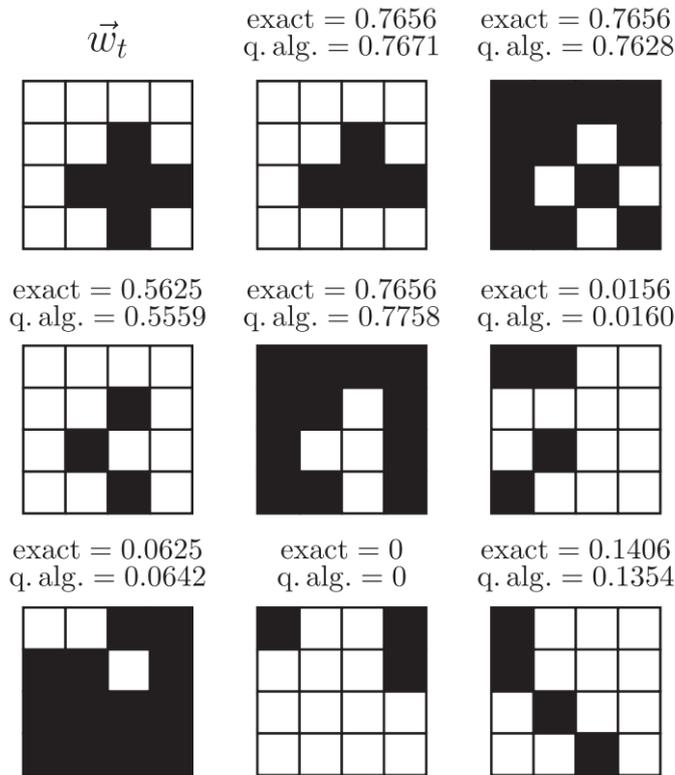
Expandindo o número de qubits para 4, o tamanho dos vetores de entrada e peso passa a ser $2^4 = 16$, enquanto o número de combinações de \vec{i} e \vec{w} , passa a 2^{32} . Esse aumento exponencial evidencia mais uma vez o mérito da computação quântica. Dessa forma, afim de testar a capacidade de escalabilidade do algoritmo, foram realizadas simulações para

¹ <https://ibm.biz/qiskit-tenerife>

² Biblioteca que fornece um conjunto de ferramentas de código aberto desenvolvido pela IBM para programação, simulação e execução de circuitos quânticos em computadores quânticos reais e simulados.

$N = 4$. Para fins de exemplo, foi escolhido um padrão simples de \vec{w}_t único em formato de cruz, representado por uma imagem de 4×4 *pixels*, como mostra a Figura 14.

Figura 14 – Simulação para vetores \vec{i} e \vec{w} com 16 elementos.



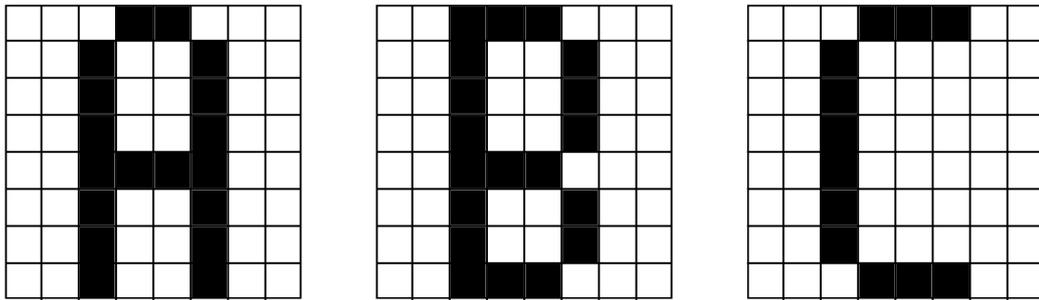
Fonte: (TACCHINO et al., 2019, p. 5)

A primeira imagem, representa a forma escolhida para \vec{w}_t e as demais, representam diferentes entradas analisadas. Logo acima de cada entrada, estão apresentados os valores de $|c_{m-1}|^2$ obtidos, em que *exact* apresenta o valor obtido numericamente, através do cálculo utilizando álgebra linear, e *q. alg.* apresenta o valor médio obtido através da simulação do Perceptron quântico com 8192 repetições. Avaliando, os resultados obtidos através das simulações concordam com os resultados calculados numericamente, de maneira com que a diferença máxima entre eles, para os exemplos acima, é de apenas 0,0102.

Aumentando ainda mais o número de qubits, um terceiro experimento foi feito. Utilizando $N = 6$ qubits, ou seja, vetores \vec{i} e \vec{w} de tamanho $2^6 = 64$, foi testada a capacidade de o algoritmo proposto reconhecer as letras do alfabeto. Para esse fim, cada letra foi representada através de uma imagem de 8×8 *pixels*, seguindo a mesma ideia das representações anteriores. A Figura 15 mostra como as letras **A**, **B** e **C** podem ser representadas por imagens de 8×8 *pixels*, lembrando que, os *pixels* em branco são representados por 1 e os pretos são representados por -1 .

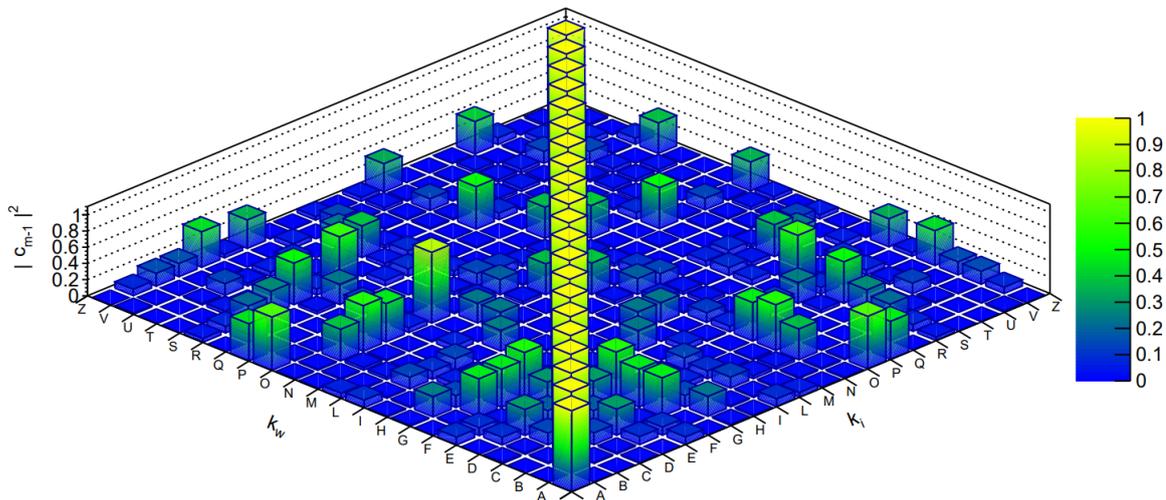
O algoritmo foi executado para todas as possíveis combinações de letras. O resultado dessas simulações pode ser visto na Figura 16.

Figura 15 – Exemplo de representação das letras A, B e C através de imagens de 8×8 pixels.



Fonte: Produzida pelos autores.

Figura 16 – Resultado obtido através da execução do Perceptron para todas as possíveis combinações de letras.



Fonte: (BON, 2019, p. 14)

Novamente, os resultados concordam com o esperado numericamente. O Perceptron é perfeitamente ativado, ou seja, $|c_{m-1}|^2 = 1$, para valores iguais de \vec{i} e \vec{w} , e 0 ou muito próximo de 0 para a maioria dos outros casos. É interessante observar o fenômeno que ocorre para letras parecidas, como por exemplo, **G** e **O**. O histograma apresenta algumas variações nesses casos que vão de 0, 1 a 0,87, o que faz com que, para uma análise ideal de se a letra dada na entrada é refletida no vetor de peso, seja feita uma iteração sob todas as letras representadas pelo vetor \vec{w} e, posteriormente, escolher a que apresenta maior valor de $|c_{m-1}|^2$ como resultado. Esse tipo de resultado, nos deixa ainda mais otimistas quanto a esse tipo de algoritmo, pois, com um único neurônio Perceptron, é plausível fazer o reconhecimento de palavras e até mesmo frases. Coisas que não são possíveis com o Perceptron quântico devida à limitação a dados linearmente separáveis.

4.0.2 Classificando cores de 8 bits

A representação de cores em 8 bits é um método antigo e simples de codificar cores em sistemas de computação. Nessa representação, cada pixel de uma imagem é dado por um valor de 8 bits, o que indica que, com essa modelagem obtém-se até $2^8 = 256$ cores diferentes. No sistema RGB, a distribuição de bits é dada por $\{R, R, R, G, G, G, B, B\}$, e é seguindo esse formato que trabalharemos nesta experimentação.

Como vimos, utilizando a computação quântica, para representar cadeias de 8 bits, precisamos de apenas 3 qubits. Dessa maneira, a classificação de cores é um experimento ideal para mostrar de forma didática como o algoritmo do Perceptron proposto funciona em seu passo a passo e quais resultados obtém-se através dessa simulação. Para este fim, vamos analisar as três cores primitivas do sistema RGB: vermelho, verde e azul, e também o roxo, que é a composição do azul e vermelho. Portanto, faremos a seguir a classificação para as seguintes combinações de \vec{i} e \vec{w} :

$$\begin{aligned}\{\vec{i}, \vec{w}\} &= \{Vermelho, Vermelho\} \\ \{\vec{i}, \vec{w}\} &= \{Vermelho, Verde\} \\ \{\vec{i}, \vec{w}\} &= \{Vermelho, Azul\} \\ \{\vec{i}, \vec{w}\} &= \{Verde, Azul\} \\ \{\vec{i}, \vec{w}\} &= \{Verde, Roxo\}\end{aligned}$$

em que, as representações binárias em cadeias de 8 bits para os valores *Vermelho*, *Verde*, *Azul* e *Roxo*, são:

$$\begin{aligned}\{Vermelho\} &= \{1, 1, 1, 0, 0, 0, 0, 0\} \equiv \{-1, -1, -1, 1, 1, 1, 1, 1\} \\ \{Verde\} &= \{0, 0, 0, 1, 1, 1, 0, 0\} \equiv \{1, 1, 1, -1, -1, -1, 1, 1\} \\ \{Azul\} &= \{0, 0, 0, 0, 0, 0, 1, 1\} \equiv \{1, 1, 1, 1, 1, 1, -1, -1\} \\ \{Roxo\} &= \{1, 1, 1, 0, 0, 0, 1, 1\} \equiv \{-1, -1, -1, 1, 1, 1, -1, -1\}\end{aligned}\tag{4.1}$$

1. Caso 1: $\vec{i} = \{Vermelho\}$, $\vec{w} = \{Vermelho\}$

O estado $|\psi_i\rangle$ do vetor que representa a cor vermelha será dado por:

$$\begin{aligned}|\psi_i\rangle &= \frac{1}{\sqrt{m}} \sum_{j=0}^{m-1} i_j |j\rangle \\ |\psi_i\rangle &= \frac{1}{\sqrt{2^3}} \sum_{j=0}^{2^3-1} i_j |j\rangle \\ |\psi_i\rangle &= \frac{1}{\sqrt{8}} \cdot (-|000\rangle - |001\rangle - |010\rangle + |011\rangle + |100\rangle + |101\rangle + |110\rangle + |111\rangle)\end{aligned}\tag{4.2}$$

Logo, o estado $|\psi_w\rangle$ será idêntico, dado por:

$$\begin{aligned} |\psi_w\rangle &= \frac{1}{\sqrt{m}} \sum_{j=0}^{m-1} w_j |j\rangle \\ |\psi_w\rangle &= \frac{1}{\sqrt{2^3}} \sum_{j=0}^{2^3-1} w_j |j\rangle \\ |\psi_w\rangle &= \frac{1}{\sqrt{8}} \cdot (-|000\rangle - |001\rangle - |010\rangle + |011\rangle + |100\rangle + |101\rangle + |110\rangle + |111\rangle) \end{aligned} \quad (4.3)$$

Voltando à descrição do Perceptron dada na seção 3, temos que o produto interno entre os vetores \vec{i} e \vec{w} , dado que são iguais, portanto o ângulo entre eles é 0, será:

$$\langle \psi_i | \psi_w \rangle = \cos 0 = 1 \quad (4.4)$$

Portanto, o retorno para este caso, será “Perceptron ativado” com $|c_{m-1}|^2 = 1$. Ou seja, o algoritmo reconhece a cor vermelha com precisão.

2. Caso 2: $\vec{i} = \{\text{Vermelho}\}$, $\vec{w} = \{\text{Verde}\}$

O estado $|\psi_i\rangle$ do vetor que representa a cor vermelha será dado pela equação 4.2, ou seja:

$$|\psi_i\rangle = \frac{1}{\sqrt{8}} \cdot (-|000\rangle - |001\rangle - |010\rangle + |011\rangle + |100\rangle + |101\rangle + |110\rangle + |111\rangle)$$

Já o estado $|\psi_w\rangle$, correspondente à cor verde, será dado por:

$$|\psi_w\rangle = \frac{1}{\sqrt{8}} \cdot (|000\rangle + |001\rangle + |010\rangle - |011\rangle - |100\rangle - |101\rangle + |110\rangle + |111\rangle) \quad (4.5)$$

O produto interno entre os vetores \vec{i} e \vec{w} será:

$$\langle \psi_i | \psi_w \rangle = \left(-\frac{4}{8}\right) = -0,5 \quad (4.6)$$

Portanto, o retorno para este caso, será o Perceptron não ativo, com $|c_{m-1}|^2 = 0,25$. Ou seja, dada a cor de entrada vermelha, o algoritmo responde corretamente caso a comparação seja com a cor verde, desde que qualquer limiar maior que 0,25 seja aplicado.

3. Caso 3: $\vec{i} = \{\text{Vermelho}\}$, $\vec{w} = \{\text{Azul}\}$

Novamente, o estado $|\psi_i\rangle$ do vetor que representa a cor vermelha será dado pela equação 4.2. Já o estado $|\psi_w\rangle$, correspondente à cor azul, será dado por:

$$|\psi_w\rangle = \frac{1}{\sqrt{8}} \cdot (|000\rangle + |001\rangle + |010\rangle + |011\rangle + |100\rangle + |101\rangle - |110\rangle - |111\rangle) \quad (4.7)$$

O produto interno entre os vetores \vec{i} e \vec{w} será:

$$\langle \psi_i | \psi_w \rangle = \left(-\frac{2}{8} \right) = -0,25 \quad (4.8)$$

O retorno para este caso indica o Perceptron não ativo, com $|c_{m-1}|^2 = 0,0625$. Dada a cor de entrada vermelha, o algoritmo responde corretamente caso a comparação seja com a cor azul, desde que qualquer limiar maior que 0,0625 seja aplicado.

4. Caso 4: $\vec{i} = \{Verde\}$, $\vec{w} = \{Azul\}$

De acordo os casos anteriores, a representação da cor verde será dada pela equação 2, enquanto a representação da cor azul será dada por 2. O produto interno entre eles, será:

$$\langle \psi_i | \psi_w \rangle = \left(-\frac{4}{8} \right) = -0,5 \quad (4.9)$$

Portanto, Perceptron não ativo com $|c_{m-1}|^2 = 0,25$.

5. Caso 5: $\vec{i} = \{Verde\}$, $\vec{w} = \{Roxo\}$

O estado $|\psi_i\rangle$ do vetor que representa a cor verde será dado pela equação 2. Já o estado $|\psi_w\rangle$, correspondente à cor roxa, será dado por:

$$|\psi_w\rangle = \frac{1}{\sqrt{8}} \cdot (-|000\rangle - |001\rangle - |010\rangle + |011\rangle + |100\rangle + |101\rangle - |110\rangle - |111\rangle) \quad (4.10)$$

Olhando para as componentes dos estados $|\psi_i\rangle$ e $|\psi_w\rangle$, é notável que são vetores ortogonais. Uma vez que o ângulo entre eles é 90° , produto interno será:

$$\langle \psi_i | \psi_w \rangle = \cos 90 = 0 \quad (4.11)$$

O retorno para este caso, será o Perceptron perfeitamente não ativo, com $|c_{m-1}|^2 = 0$.

Analisando os resultados obtidos através da simulação para esses cinco casos, é perceptível a possibilidade de classificação de qualquer cor representada por 8 bits, além da possibilidade de se estender sistemas com mais cores. Outro ponto que fica evidente, é que quanto mais diferentes entre si \vec{i} e \vec{w} forem, menor será a chance de se encontrar um “Perceptron ativo”. O efeito inverso, pode ser um problema para conjuntos que possuam dados muito semelhantes, mas pode ser facilmente corrigido com a aplicação de um limiar. A principal vantagem da utilização da computação quântica fica explícita, uma vez que para a representação de qualquer combinação de 2^8 cores, foi necessário utilizar apenas 3 qubits. Outro resultado que também é visível nas análises da seção 4.0.1, é que este Perceptron, é capaz de superar o Perceptron clássico, uma vez que um único neurônio é capaz de classificar dados independentemente da não-linearidade.

5 Conclusão

A busca pela compreensão dos fundamentos da Computação Quântica é importante por ser um tema em evidência, com implicações para o futuro da tecnologia. Neste trabalho, abordamos alguns aspectos essenciais que são a base para a computação quântica e posteriormente, especificamos os estudos em torno das Redes Neurais Artificiais. Graças ao princípio de superposição, propriedade que faz com que a computação quântica tenha um grau de paralelismo natural, há uma grande promessa envolvendo a IA e o aprendizado de máquina. Além disso, a impossibilidade de cópia e a capacidade de troca de informações permitida através do emaranhamento abrem portas para uma criptografia mais segura e eficaz.

O benefício da computação quântica para a Inteligência Artificial, como foi visto, surge através do paralelismo intrínseco como propriedade dos sistemas quânticos e da capacidade exponencial de computação de dados. Neste sentido, o desenvolvimento deste trabalho, ocorreu principalmente no intuito de entender o potencial da computação quântica no que diz respeito à essa interseção. Para esse fim, foi realizado um estudo sobre um modelo de Perceptron quântico. Os resultados acerca dos experimentos envolvendo esse algoritmo, evidenciaram o ganho exponencial ao se utilizar os modelos quânticos para fins de análise e classificação de dados: com apenas N qubits, somos capazes de representar 2^N estados diferentes. Além disso, os resultados mostraram que o modelo é capaz de prever corretamente os padrões estipulados nas simulações. Ao analisar os resultados obtidos nos trabalhos em que este estudo foi fundamentado, é possível concluir que, como esperado, as simulações em processadores quânticos, concordam com os experimentos matemáticos. Isso nos permitiu realizar a simulação matemática de classificação de cores de 8 bits utilizando o Perceptron. Com esse objetivo, analisamos as respostas obtidas classificando as cores vermelha, verde, azul e roxo, utilizando o sistema RGB. Os cálculos mostraram que um único Perceptron é capaz de responder corretamente a qual cor uma entrada corresponde, de maneira com que, aplicado um limiar de 0,25, o algoritmo responde corretamente para todas as combinações propostas de cores.

Um comportamento interessante também foi observado: quanto maior a diferença entre a entrada analisada e o objeto de comparação, menor o valor da probabilidade medida. Uma vez que o comportamento inverso também é válido, surge a necessidade de análise do conjunto de dados e da aplicação de um limiar que capacite a classificação entre objetos que apresentem alta similaridade.

Um dos resultados mais empolgantes dessa pesquisa, é a evidência da não existência de limitação a respeito da necessidade de linearidade do conjunto de dados. O que possibilita

que apenas um neurônio do Perceptron quântico, classifique padrões que o clássico seria incapaz. Esse estudo abre precedentes para posteriores análises a respeito de padrões ainda mais complexos. Além disso, mantém a esperança de que, por mais que ainda esteja restrita devido à dificuldade de implementação e altos custos relacionados, a computação quântica traga a solução para problemas que permanecem por muito tempo intratáveis na computação clássica.

Referências

- ARUTE, F. et al. Quantum supremacy using a programmable superconducting processor. *Nature*, Nature Publishing Group, v. 574, n. 7779, p. 505–510, 2019. Citado 2 vezes nas páginas 14 e 16.
- BARENCO, A. et al. Elementary gates for quantum computation. *Physical review A*, APS, v. 52, n. 5, p. 3457, 1995. Citado na página 23.
- BON, R. Quantum algorithm for the implementation of a perceptron. 2019. Citado 4 vezes nas páginas 16, 37, 43 e 46.
- COHEN-TANNOUJDI, C.; LALOE, F.; DIU, B. Quantum mechanics. Wiley-Vch, 2019. Citado na página 18.
- DEUTSCH, D.; JOZSA, R. Rapid solution of problems by quantum computation. *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences*, The Royal Society London, v. 439, n. 1907, p. 553–558, 1992. Citado na página 29.
- DJORDJEVIC, I. Chapter 2 - quantum mechanics fundamentals. In: DJORDJEVIC, I. (Ed.). *Quantum Information Processing and Quantum Error Correction*. Oxford: Academic Press, 2012. p. 29–89. ISBN 978-0-12-385491-9. Disponível em: <<https://www.sciencedirect.com/science/article/pii/B9780123854919000022>>. Citado na página 25.
- JAEGER, G. Bell gems: the bell basis generalized. *Physics Letters A*, Elsevier, v. 329, n. 6, p. 425–429, 2004. Citado na página 22.
- LEWENSTEIN, M. Quantum perceptrons. *Journal of Modern Optics*, Taylor & Francis, v. 41, n. 12, p. 2491–2501, 1994. Citado na página 37.
- MCCLULLOCH, W. S.; PITTS, W. A logical calculus of the ideas immanent in neurons activity. *Bulletin of mathematical biophysics*, v. 5, n. 115-133, p. 10, 1943. Citado na página 36.
- NIELSEN, M. A.; CHUANG, I. L. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. [S.l.]: Cambridge University Press, 2010. Citado na página 24.
- ROSENBLATT, F. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, American Psychological Association, v. 65, n. 6, p. 386, 1958. Citado na página 14.
- TACCHINO, F. et al. An artificial neuron implemented on an actual quantum processor. *npj Quantum Information*, Nature Publishing Group, v. 5, n. 1, p. 26, 2019. Citado 7 vezes nas páginas 37, 38, 40, 41, 43, 44 e 45.
- WIEBE, N.; KAPOOR, A.; SVORE, K. M. Quantum perceptron models. *arXiv preprint arXiv:1602.04799*, 2016. Citado na página 38.
- ZHOU, R.; QIN, L.; JIANG, N. Quantum perceptron network. In: SPRINGER. *Artificial Neural Networks–ICANN 2006: 16th International Conference, Athens, Greece, September 10-14, 2006. Proceedings, Part I 16*. [S.l.], 2006. p. 651–657. Citado na página 37.