SIMULAÇÃO QUÂNTICA: Uma análise comparativa entre algoritmos clássicos e quânticos[1]

Guilherme Moreira de Carvalho[2]
Tales Argolo Jesus[3]

Submetido em: 03/06/2026 | Avaliado em: 12/06/2026

RESUMO

A área da computação passou por uma transformação extraordinária nas últimas décadas. Algoritmos de busca, antes projetados para resolver problemas relativamente simples e bem definidos, agora operam em ambientes complexos e ricos em dados. Essa mudança reflete uma tendência mais ampla: a evolução dos paradigmas da computação tem remodelado repetidamente o cenário da inovação tecnológica. Neste artigo, paralelos temáticos e métricas de desempenho para implementação e simulação em diferentes arquiteturas são definidos por meio de uma comparação entre algoritmos de busca clássicos (linear; binária) e quânticos (Grover). Os resultados validam a complexidade teórica de cada algoritmo e evidenciam os cenários adequados para sua aplicação, destacando o custo computacional extra de operações como ordenação na busca binária e conversão na busca quântica. Tal análise oferece um ponto de partida para essa área de pesquisa para estudantes de Ciências e Matemática.

Palavras-chave: Computação Quântica. Algoritmos. Paradigmas de computação. Algoritmo de Grover.

ABSTRACT

The field of computation has undergone extraordinary transformation over the past several decades. Search algorithms, once designed to address relatively narrow and well-defined problems, now operate in complex, data-rich environments. This shift reflects a broader trend: changes in computing paradigms have repeatedly reshaped the landscape of technological innovation. In this paper, thematic parallels and performance metrics for implementation and simulation in different architectures are defined through a comparison of classical (Linear; Binary) and quantum (Grover) search algorithms. The results validate the theoretical complexity of each algorithm and highlight the appropriate scenarios for its application, emphasizing the extra computational cost of operations such as sorting in binary search and conversion in

[1] Artigo científico elaborado para o trabalho de conclusão do curso de Engenharia de Computação do campus Nova Gameleira do Centro Federal de Educação Tecnológica de Minas Gerais (CEFET-MG.)

[2] Graduando em Engenharia de Computação no campus Nova Gameleira do Centro Federal de Educação Tecnológica de Minas Gerais (CEFET-MG.).

[3] Orientador do trabalho. Docente do Departamento de Computação do campus Nova Gameleira do Centro Federal de Educação Tecnológica de Minas Gerais (CEFET-MG.). tales@cefetmg.br

quantum search. Such analysis offers an entry point to this research area for students in Sciences and Mathematics.

Keywords: Quantum Computing. Algorithms. Computing paradigms. Grover's algorithm.

1 INTRODUÇÃO

A evolução da computação é marcada por avanços contínuos no desenvolvimento de algoritmos e nas capacidades de hardware, cada um moldando e possibilitando o outro. Entre as inovações mais influentes nessa trajetória estão os algoritmos de busca, que surgiram no início da ciência da computação como ferramentas essenciais para a resolução de problemas, o planejamento de rotas e a recuperação de dados (BULLYNCK, 2015). Métodos fundamentais como o algoritmo de Dijkstra e abordagens heurísticas posteriores como o $A*$ estabeleceram princípios que continuam a sustentar aplicações modernas. À medida que os conjuntos de dados e as tarefas computacionais se tornaram mais complexos, a demanda por mecanismos de busca mais rápidos e eficientes impulsionou tanto o progresso teórico quanto a inovação prática (DAHL *et al.*, 1972).

Esse avanço algorítmico foi acompanhado por um crescimento sem precedentes no hardware, descrito pela Lei de Moore ao observar que o número de transistores em circuitos integrados dobra aproximadamente a cada dois anos. Por décadas, essa escalabilidade exponencial proporcionou melhorias contínuas de desempenho, permitindo que algoritmos de busca cada vez mais sofisticados operassem em tempo real e em espaços de problemas cada vez maiores. No entanto, à medida que a miniaturização de transistores se aproxima de seus limites físicos, a desaceleração da Lei de Moore promoveu uma mudança em direção a arquiteturas alternativas e modelos de computação heterogêneos (HENNESSY; PATTERSON, 2012).

Nesse contexto, a computação quântica emergiu como um paradigma promissor, capaz de redefinir os limites computacionais, oferecendo novas possibilidades algorítmicas baseadas nos princípios da mecânica quântica. Ao explorar a superposição e o emaranhamento, os sistemas quânticos possibilitam estratégias conceitualmente novas, incluindo ganhos quadráticos em buscas não estruturadas por meio do algoritmo de Grover (GROVER, 1996). Embora os dispositivos quânticos contemporâneos ainda sejam limitados por ruído e escala, os esforços contínuos de organizações como IBM e Google impulsionam a área em direção a aplicações práticas que complementam ou superam as técnicas clássicas de pesquisa em otimização, criptografia e análise de grafos (WONG, 2022).

A relação evolutiva entre a computação clássica e a computação quântica, bem como a ainda incipiente difusão dos fundamentos dessa na base curricular da engenharia viabilizam a construção de paralelos conceituais para fins didáticos. Além disso, a definição de métricas de desempenho para a implementação e simulação em diferentes arquiteturas explicitam os avanços tecnológicos alcançados por ambas. Os algoritmos de busca constituem uma escolha ótima para comparar os paradigmas de computação porque representam um problema fundamental, amplamente estudado e com limites de complexidade bem estabelecidos em ambas as abordagens.

Neste artigo, é realizada uma implementação do algoritmo quântico de busca de Grover a fim de comparar sua complexidade e desempenho com os algoritmos de busca clássicos linear e binário (CORMEN *et al.*, 2009). Esse estudo conduz uma abordagem

introdutória, porém rigorosa e prática, enfatizando a natureza evolutiva dos algoritmos e paradigmas computacionais. Ao comparar sistematicamente algoritmos de busca clássicos com sua contraparte quântica e ao empregar ferramentas acessíveis, este trabalho preenche a lacuna entre os avanços teóricos em computação quântica e a compreensão prática.

2 REFERENCIAL TEÓRICO

Este seção apresenta os conceitos fundamentais que sustentam esta pesquisa, com foco nos algoritmos de busca e na computação quântica. São discutidos os princípios de qubits, emaranhamento quântico, portas lógicas quânticas e circuitos lógicos quânticos.

2.1 Algoritmos de Busca

A busca linear é o método mais simples para encontrar um valor em uma lista: ela verifica cada elemento um após o outro, começando do início e avançando até encontrar o alvo ou chegar ao final. A cada passo, o algoritmo compara o elemento atual com o valor procurado; se houver correspondência, o algoritmo para e retorna a posição; caso contrário, continua para o próximo elemento. Como pode ser necessário examinar todos os itens, seu tempo de execução no pior caso é proporcional ao tamanho da lista, tornando-a prática para conjuntos de dados pequenos ou não ordenados, mas ineficiente para conjuntos grandes.

A busca binária é um algoritmo eficiente para encontrar um valor alvo em uma lista ordenada, dividindo repetidamente o intervalo de busca ao meio. Em vez de verificar cada elemento sequencialmente, ela compara o alvo com o elemento do meio: se o alvo for igual ao valor do meio, a busca termina; se for menor, o algoritmo continua a busca na metade esquerda; se for maior, busca na metade direita. Como cada passo reduz o espaço de busca ao meio, a busca binária tem complexidade logarítmica, tornando-a muito mais rápida que a busca linear para conjuntos de dados grandes e ordenados, mas inviável em listas que não estejam previamente ordenadas.

O algoritmo de Grover é um método de busca quântica que oferece uma vantagem de velocidade sobre as técnicas clássicas, utilizando superposição quântica e amplificação de amplitude para localizar um item alvo em uma lista não ordenada (NIELSEN; CHUANG, 2010). Conceitualmente, ele desempenha um papel semelhante à busca linear e binária, mas com um perfil de desempenho diferente: a busca linear verifica os elementos um a um e, portanto, apresenta complexidade computacional $O(N)$, enquanto a busca binária divide o espaço de busca pela metade repetidamente e atinge complexidade $O(\log N)$. A abordagem de Grover, no entanto, funciona com dados não ordenados como a busca linear, mas requer apenas $O(\sqrt{N})$ consultas (ver seção 2.3), proporcionando um ganho de velocidade quadrático. Embora não seja tão rápido quanto o desempenho logarítmico da busca binária, o algoritmo de Grover é notável porque supera o melhor método clássico possível para busca não estruturada, demonstrando como a computação quântica pode remodelar fundamentalmente nossas expectativas sobre os limites algorítmicos.

2.2 Qubits

Qubits (bits quânticos) são a unidade básica de informação da computação quântica, análogos aos bits da computação clássica, porém com propriedades fundamentais diferentes (SCHUMACHER, 1995). Na computação clássica, um bit pode assumir apenas um de dois

valores: 0 ou 1. Esses valores podem ser expressos como dois níveis de tensão ou corrente em um circuito ou duas posições de interruptores físicos (WONG, 2022), por exemplo. Já um qubit é descrito por um estado quântico, podendo estar em superposição dos estados $|0\rangle$ e $|1\rangle$ simultaneamente. Matematicamente, o estado de um qubit pode ser representado como:

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle = \alpha \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \beta \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$
$$= \begin{pmatrix} \alpha \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ \beta \end{pmatrix} = \begin{pmatrix} \alpha \\ \beta \end{pmatrix},$$

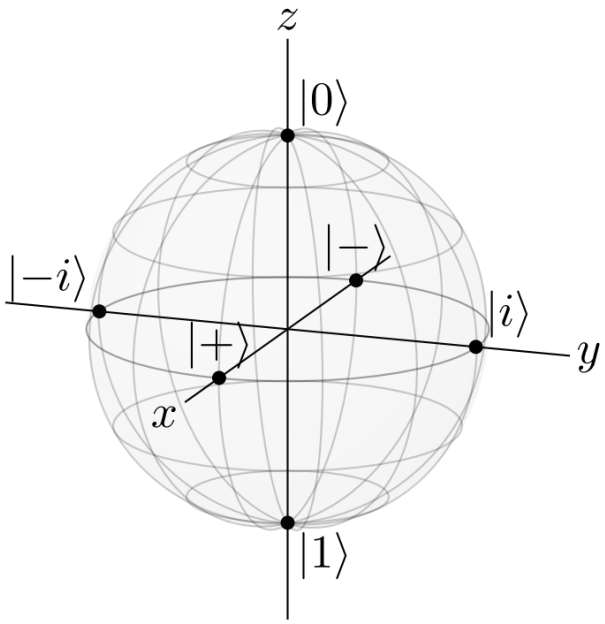
sendo α e β números complexos chamados amplitudes de probabilidade que satisfazem a condição $|\alpha|^2 + |\beta|^2 = 1$. Ao realizar uma medição, o qubit colapsa para o estado $|0\rangle$ com probabilidade $|\alpha^2|$ ou para $|1\rangle$ com probabilidade $|\beta|^2$ (GRIFFITHS; SCHROETER, 2018).

A figura 1 ilustra geometricamente o espaço de estados de um qubit na Esfera de Bloch (FEYNMAN *et al.*, 1957). Os pólos norte e sul correspondem aos estados $|0\rangle$ e $|1\rangle$ respectivamente. Os demais pontos podem ser descritos como uma superposição desses.

$$|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$
$$|-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$$
$$|i\rangle = \frac{1}{\sqrt{2}}(|0\rangle + i\,|1\rangle)$$
$$|-i\rangle = \frac{1}{\sqrt{2}}(|0\rangle - i\,|1\rangle)$$

.

Figura 1 – Representação dos estados notáveis de um qubit na esfera de Bloch.



Fonte: Wong (2022)

Além da superposição, qubits podem apresentar entrelaçamento quântico, um fenômeno no qual o estado de um qubit depende do estado de outro, mesmo quando separados fisicamente. Esse comportamento não possui equivalente clássico e é essencial para o funcionamento de algoritmos quânticos, como o algoritmo de Grover.

Fisicamente, qubits podem ser implementados de diferentes formas, como estados de spins de elétrons, níveis de energia de átomos, fótons ou circuitos supercondutores (NISBET-JONES *et al.*, 2013). Independentemente da implementação, o conceito de qubit permite explorar superposição, interferência e entrelaçamento, possibilitando vantagens computacionais em relação aos sistemas clássicos.

2.3 Portas Quânticas

Portas quânticas são componentes que realizam as operações fundamentais da computação quântica manipulando o estado dos qubits durante a execução de um algoritmo. Elas desempenham um papel equivalente ao das portas lógicas (AND, OR, NOT) (TOCCI *et al.*, 2011) na computação clássica, porém obedecem às leis da mecânica quântica.

Do ponto de vista teórico, portas quânticas são representadas por operações unitárias, ou seja, transformações matemáticas reversíveis que preservam a norma do estado quântico. Essas operações atuam sobre um ou mais qubits, alterando suas amplitudes de probabilidade e fases, sem realizar medição. A reversibilidade é uma característica essencial, pois portas lógicas clássicas reversíveis, cujo sinal de entrada pode ser recuperado a partir da saída, são portas quânticas válidas e a evolução de sistemas quânticos isolados, como o oráculo (ver seção 2.3), é determinística e reversível.

Entre as portas quânticas de um qubit, destaca-se a porta de Hadamard (H), amplamente utilizada para criar estados de superposição, transformando um qubit numa combinação equilibrada de $|0\rangle$ e $|1\rangle$. Outras portas importantes incluem a porta de Pauli-X, que atua de forma semelhante ao operador NOT clássico, além das portas de Pauli-Y e Pauli-Z, responsáveis por rotações nos respectivos eixos na Esfera de Bloch (FILATOV; AUZINSH, 2024).

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

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

As portas quânticas de múltiplos qubits, como a porta CNOT ou Controlled-NOT (Figura 2), são fundamentais para gerar entrelaçamento quântico. A CNOT altera o estado de um qubit alvo dependendo do estado de um qubit de controle, permitindo criar correlações quânticas que não podem ser reproduzidas por sistemas clássicos.

$$CNOT|00\rangle = |00\rangle$$

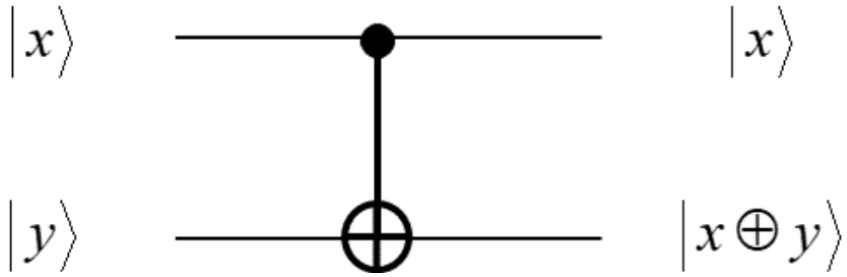
$$CNOT|01\rangle = |01\rangle$$

$$CNOT|10\rangle = |11\rangle$$

$$CNOT|11\rangle = |10\rangle$$

Conjuntos organizados de portas quânticas formam os circuitos quânticos, que representam a implementação prática dos algoritmos quânticos. A figura 3 ilustra o circuito correspondente ao algoritmo de Grover (ver seção 2.3), no qual portas de Hadamard são

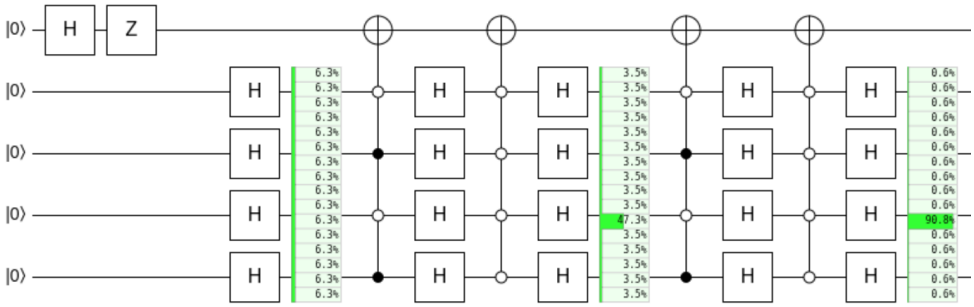
Figura 2 – Porta quântica CNOT de dois qubits.



Fonte: Quantum Computing Stack Exchange

usadas para criar superposição, enquanto combinações específicas de portas Pauli-X e de controle atuam no sistema. Simuladores visuais de circuitos quânticos, como o Quirk, tem como principal objetivo facilitar o aprendizado de computação quântica por meio da construção gráfica de circuitos, sem necessidade de programação.

Figura 3 – Simulação do circuito do algoritmo de Grover para 4 qubits na ferramenta *Quirk*.



Disponível em: <https://algassert.com/quirk>

2.4 Algoritmo de Grover

O algoritmo de Grover é um dos algoritmos mais conhecidos e importantes na computação quântica. Ele foi desenvolvido por Lov Grover em 1996 para resolver o problema de busca em uma base de dados não ordenada (GROVER, 1996). Em vez de procurar um elemento específico de forma sequencial (como na busca linear clássica), o algoritmo de Grover utiliza a superposição dos estados para consultar todos os itens da lista conjuntamente. Isso oferece uma vantagem quadrática em relação aos algoritmos clássicos.

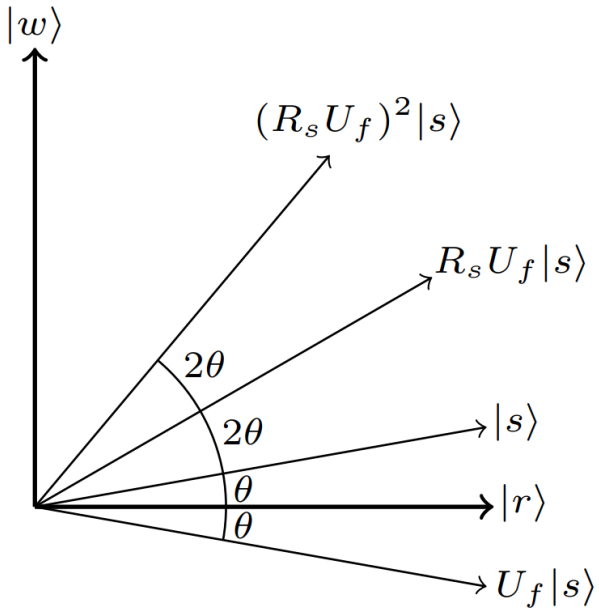
Seu funcionamento é estruturado em quatro etapas principais: superposição, oráculo, amplificação e medição. Inicialmente, os qubits são preparados em uma superposição balanceada de todos os N estados possíveis, obtida pela aplicação da porta de Hadamard a cada um deles:

$$|s\rangle = H\,|x^n\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle\,.$$

O estado de superposição $|s\rangle$ pode ser decomposto em uma combinação do estado alvo $|w\rangle$ e os restantes $|r\rangle$ presentes no sistema, permitindo uma visualização geométrica

simples do problema, já que esses são perpendiculares entre si (Figura 4). Essa interpretação é fundamental para o algoritmo de Grover, pois a quantidade de iterações ótima é tal que o estado de superposição é rotacionado em múltiplos do ângulo θ que forma com os não-alvo até se afastar suficiente deles, percorrendo todo o arco perpendicular e atingindo o alvo.

Figura 4 – Representação gráfica do algoritmo de Grover.



Fonte: Wong (2022)

$$
\begin{aligned}
|s\rangle &= \frac{1}{\sqrt{N}}(|w\rangle + \sum_{i \neq w} |i\rangle) \\
&= \frac{1}{\sqrt{N}} |w\rangle + \frac{1}{\sqrt{N}} \sum_{i \neq w} |i\rangle \\
&= \frac{1}{\sqrt{N}} |w\rangle + \sqrt{\frac{N-1}{N}} \frac{1}{\sqrt{N-1}} \sum_{i \neq w} |i\rangle \\
&= \frac{1}{\sqrt{N}} |w\rangle + \sqrt{\frac{N-1}{N}} |r\rangle \\
&\sin\theta |w\rangle + \cos\theta |r\rangle ,
\end{aligned}
$$

sendo θ definido pela razão trigonométrica (VENTURI, 2015):

$$\sin\theta = \frac{1}{\sqrt{N}}, \cos\theta = \sqrt{\frac{N-1}{N}}.$$

Em seguida, o oráculo é responsável por marcar o estado correspondente à solução por meio de uma inversão de fase, aplicada exclusivamente ao estado alvo:

$$U_f |x\rangle = (-1)^{f(x)} |x\rangle ,$$

em que $f(x) = 1$ se x é o alvo e $f(x) = 0$ caso contrário. Entretanto, a fase negativa não é suficiente para gerar o estado desejado, uma vez que não altera a distribuição de probabilidade dada pela norma quadrática das amplitudes. Assim, o oráculo prepara o sistema para a próxima etapa.

$$\begin{aligned} U_f |s\rangle &= (-1)^1 \sin\theta |w\rangle + (-1)^0 \cos\theta |r\rangle \\ &= -\sin\theta |w\rangle + \cos\theta |r\rangle . \end{aligned}$$

A amplificação de amplitude no algoritmo de Grover é o mecanismo responsável por aumentar progressivamente a probabilidade de se medir o estado correspondente à solução correta e, consequentemente, reduzir a probabilidade dos demais. Ela ocorre por meio do operador de difusão R_s, que realiza uma transformação conhecida como inversão sobre a média e reflete as amplitudes de todos os estados em torno do valor médio do sistema - representado pelo estado de superposição $|s\rangle$. Assim, a amplitude do estado marcado, inicialmente negativa e menor que as demais, passa a ficar acima da média.

Cada aplicação conjunta do oráculo e do operador de difusão constitui uma iteração de Grover, e a repetição controlada dessas iterações intensifica a interferência construtiva associada ao estado solução. Em espaços de busca grandes em que N é um valor relativamente alto, o seno de θ é aproximadamente o próprio valor de θ. Então, uma aplicação consecutiva do oráculo U_f e do refletor R_s, caracterizando uma iteração de Grover, rotaciona o estado $|s\rangle$ em direção ao estado $|w\rangle$ em 2θ e amplifica naquele o componente relacionado a este, resultando em uma alta probabilidade de obtenção da solução quando a medição é realizada.

$$\begin{aligned} Repetições &\approx \frac{\pi/2}{2\theta} = \frac{\pi}{4} \cdot \frac{1}{\theta} \\ &= \frac{\pi}{4}\sqrt{N} \end{aligned}$$

3 TRABALHOS RELACIONADOS

Existem na literatura trabalhos focados em desenvolver algoritmos e projetar circuitos quânticos factíveis análogos às soluções já encontradas e difundidas pela computação clássica. Por exemplo, o trabalho realizado por Ray (2014) apresenta uma adaptação quântica do algoritmo clássico de Dijkstra, um dos algoritmos mais importantes para resolução de problemas de caminhos mínimos em grafos. O trabalho parte da ideia de que a etapa mais custosa do algoritmo clássico consiste na busca repetida pelo nó de menor custo, operação tradicionalmente realizada de forma linear e com complexidade elevada. Para contornar essa limitação, a autora propõe o Algoritmo Quântico de Dijkstra (QDA) e um modelo para seu circuito físico realizável que utiliza o algoritmo de Grover para acelerar a busca pelos menores valores em um conjunto não ordenado. No modelo proposto, os custos dos nós vizinhos são colocados em superposição quântica e processados por iterações de Grover, permitindo marcar estados correspondentes aos menores caminhos e reduzindo a complexidade da busca.

Por sua vez, o trabalho realizado por Qiu *et al.* (2024) vai além e oferece uma versão distribuída do algoritmo de Grover, na qual uma função booleana é dividida em diversas subfunções menores que podem ser processadas separadamente, diminuindo a profundidade dos circuitos quânticos e o número de qubits necessários para execução

em computadores quânticos da era atual: Computação Quântica de Escala Intermediária Ruidosa (NISQ). Nesse contexto, Grover atua como uma sub-rotina de busca responsável por localizar soluções dentro de cada subconjunto analisado, enquanto algoritmos auxiliares estimam previamente a quantidade de soluções possíveis em cada subfunção para aumentar a eficiência da busca.

Essas pesquisas evidenciam que o algoritmo de Grover pode atuar como sub-rotina de otimização em problemas mais complexos, reforçando a importância de aproximar abordagens clássicas com soluções quânticas, especialmente no contexto da evolução da computação quântica e da busca por algoritmos mais eficientes. Para isso, é fundamental compreender os princípios desse novo paradigma e os conceitos que sustentam o desenvolvimento desses algoritmos, estabelecendo uma base teórica para a junção entre métodos clássicos e quânticos de programação.

De maneira similar ao presente trabalho, Jesus *et al.* (2021) apresenta uma introdução didática à computação quântica para estudantes de graduação, enfatizando a necessidade de democratizar o acesso ao conhecimento desse paradigma diante do seu avanço recente e da disponibilização de plataformas acessíveis como o IBM Quantum Platform. Os autores defendem que muitos estudantes só entram em contato com o tema na pós-graduação, tornando importante a criação de abordagens educacionais mais introdutórias e práticas para o ensino desses conteúdos. Para isso, o trabalho utiliza o Quantum Information Software Development Kit (Qiskit), kit de desenvolvimento quântico da IBM baseado em Python, como principal ferramenta didática, permitindo que alunos com pouca ou nenhuma experiência em programação científica possam construir, simular e executar circuitos quânticos em computadores pessoais ou em processadores quânticos reais acessados remotamente.

Assim, o trabalho aqui proposto difere de iniciativas de pesquisa anteriores que se concentram principalmente em propor novos algoritmos quânticos ou implementações quânticas avançadas voltadas para públicos especializados. Em vez disso, este estudo proporciona uma compreensão sobre como diferentes paradigmas computacionais impactam a resolução de problemas complexos e a vantagem quântica, aproximando estudantes de um dos horizontes tecnológicos mais promissores da atualidade. Por fim, contribui não introduzindo novos métodos quânticos, mas fornecendo uma perspectiva clara, comparativa e didática que apoia o processo de aprendizagem de estudantes e recém-chegados à área.

4 METODOLOGIA

Este trabalho adota uma abordagem experimental e comparativa para analisar o desempenho de algoritmos de busca clássicos e quânticos. O objetivo é comparar a busca linear e a busca binária, implementadas em ambiente clássico, com o algoritmo de Grover, representativo do paradigma de computação quântica.

Inicialmente, os algoritmos clássicos de busca linear e busca binária são implementados utilizando a linguagem de programação Python na plataforma de computação Jupyter Notebook, escolhidas por suas simplicidade, ampla adoção acadêmica e boas ferramentas de análise de desempenho de um programa. A busca linear percorre sequencialmente os elementos do vetor até encontrar o valor desejado, enquanto a busca binária ordena o vetor antes de realizar as operações, reduzindo o espaço de busca pela metade a cada iteração.

Para o contexto quântico, é utilizado o algoritmo de Grover, que permite acelerar a busca não estruturada em bases de dados por meio da aplicação dos princípios de

superposição e interferência quântica. A implementação do algoritmo de Grover é realizada com o auxílio do framework Qiskit, uma biblioteca open source desenvolvida pela IBM para programação e simulação de circuitos quânticos. Ela oferece samplers para medir o resultado de programas quânticos executados em simuladores locais ou em backends remotos reais e ferramentas de visualização para análise desse resultado.

Os testes são feitos com vetores de dimensões distintas para avaliar o comportamento dos algoritmos em diferentes escalas, representadas pela quantidade de números em um vetor no caso clássico e pela quantidade de qubits inicializados no caso quântico. Neste ponto, deve ser considerado o alto custo computacional de simular problemas quânticos com grande espaço de estados em computadores clássicos. Para cada vetor, são realizadas buscas por posições específicas com o objetivo de analisar como a posição do elemento afeta o desempenho de cada algoritmo, já que o tempo de execução depende diretamente do índice do item procurado.

A busca por posições de elementos marcados ao invés de elementos em si é realizada devido ao caráter teórico do modelo de consulta do qual os algoritmos de busca fazem parte. O foco da análise é a relação entre complexidade e ordenação nos casos clássicos e o funcionamento do oráculo no caso quântico. Isso não reduz a relevância do trabalho, pois a amplificação da amplitude apresentada é uma sub-rotina comum em sistemas quânticos e a arquitetura servidor-cliente, difundida em aplicações clássicas, funciona como uma caixa preta em que uma parte conhece a requisição e a outra, previamente informada do alvo dessa requisição, sabe como respondê-la.

Durante os experimentos com os algoritmos clássicos, é coletada a quantidade de elementos consultados até a localização do alvo, permitindo a comparação direta entre busca linear e busca binária. No caso do algoritmo de Grover, são analisados o número de iterações (ou aplicações do operador de Grover) e a probabilidade de sucesso da medição.

Por fim, os resultados obtidos são organizados e comparados de forma qualitativa e quantitativa, destacando as diferenças conceituais e práticas entre os modelos clássico e quântico de computação. Essa metodologia permite avaliar não apenas o ganho teórico do algoritmo de Grover, mas também os desafios relacionados à sua implementação e simulação em comparação com algoritmos clássicos amplamente utilizados.

5 DESENVOLVIMENTO

O desenvolvimento deste trabalho foi dividido em duas etapas principais: a implementação dos algoritmos clássicos de busca e a implementação do algoritmo quântico de Grover. Em ambos os casos, buscou-se utilizar ferramentas acessíveis e amplamente difundidas, permitindo a reprodução dos experimentos por estudantes e pesquisadores iniciantes na área.

Apesar da principal característica desses algoritmos ser sua simplicidade de implementação, seu desempenho depende diretamente da posição do elemento procurado. No melhor caso apenas uma comparação é necessária. Durante os testes realizados neste trabalho, as buscas foram aplicadas em vetores de diferentes tamanhos para avaliar o impacto do crescimento da entrada sobre o número de comparações necessárias.

A implementação do algoritmo de Grover foi baseada nas ferramentas disponibilizadas pelo framework Qiskit, especialmente nas primitivas de execução quântica introduzidas nas versões mais recentes da biblioteca. O circuito foi construído utilizando o operador GroverOperator, responsável por encapsular as etapas de oráculo e difusão em uma única

Algorithm 1 Pseudocódigo da busca linear

```
contador ← 0
while contador ≤ tamanho do
    if lista[contador] == chave then
        devolvecontador
    else
        contador ← contador + 1
    end if
end while
```

Algorithm 2 Pseudocódigo da busca binária

```
inicio ← 0
fim ← tamanho − 1
while inicio ≤ fim do
    meio ← (inicio + fim)/2
    if lista[meio] == chave then
        devolvemeio
    end if
    if lista[meio] > chave then
        fim ← meio − 1
    else
        inicio ← meio + 1
    end if
end while
```

estrutura reutilizável. Para execução e coleta dos resultados, foi utilizada a primitiva StatevectorSampler, que permite realizar amostragens diretamente a partir do vetor de estado quântico ideal, possibilitando observar com precisão as probabilidades associadas aos estados do sistema. Além disso, a biblioteca QiskitRuntimeService foi empregada para gerenciamento de backends e integração com os serviços da plataforma IBM Quantum, permitindo compatibilidade tanto com simuladores locais quanto com computadores quânticos reais disponibilizados em nuvem. [4]

Para isso, foi realizada a etapa de transpilação, o processo de otimização e adaptação do circuito quântico para uma arquitetura específica de execução, convertendo as operações abstratas do algoritmo em um conjunto de portas compatíveis com o backend selecionado. Isso é particularmente importante em computação quântica devido à presença de ruído e decoerência, uma vez que circuitos mais profundos e com maior quantidade de portas tendem a acumular mais erros durante a execução. Assim, a transpilação contribui para reduzir os efeitos do ruído ao minimizar a profundidade do circuito e a quantidade de operações necessárias, tornando a simulação do algoritmo de Grover mais estável e aumentando a fidelidade dos resultados obtidos.

São oferecidos diferentes níveis de otimização, variando de 0 a 3, que controlam a intensidade das transformações aplicadas ao circuito. O nível 0 realiza apenas o mapeamento básico para o backend, sem otimizações significativas, enquanto os níveis superiores aplicam simplificações progressivamente mais agressivas, como cancelamento de portas equivalentes, redução da profundidade do circuito e escolha de rotas mais eficientes entre qubits. O nível

[4] <https://quantum.cloud.ibm.com/learning/en/modules/computer-science/grovers>

3, mais avançado, busca minimizar ao máximo a quantidade de operações e a profundidade total do circuito, embora exija maior tempo de compilação.

6 RESULTADOS

Esta seção apresenta e discute os principais resultados obtidos a partir da implementação e execução dos algoritmos de busca clássicos e quânticos desenvolvidos neste trabalho. Os experimentos tiveram como objetivo analisar o comportamento da busca linear, da busca binária e do algoritmo de Grover em diferentes cenários, considerando tanto métricas de desempenho quanto aspectos conceituais relacionados aos paradigmas de computação clássica e quântica.

6.1 Algoritmos de Busca Clássicos

Para um vetor com 100.000 valores distintos, a busca linear consulta todos os índices sequencialmente até encontrar a posição desejada e seu gráfico tem comportamento crescente à medida que o alvo se distancia da origem (Figura 5). Já a busca binária tem valor mínimo de consultas no índice médio do vetor e aumenta ao passo que avança para o início ou fim, mas seu máximo nunca ultrapassa o logaritmo na base 2 do tamanho do vetor ($\log_2 100.000 = 16,60964$). Esse resultado é esperado dadas as complexidades proporcionais e logarítimicas respectivas desses algoritmos.

Figura 5 – Comparação da quantidade de iterações entre os algoritmos de busca linear e binária.

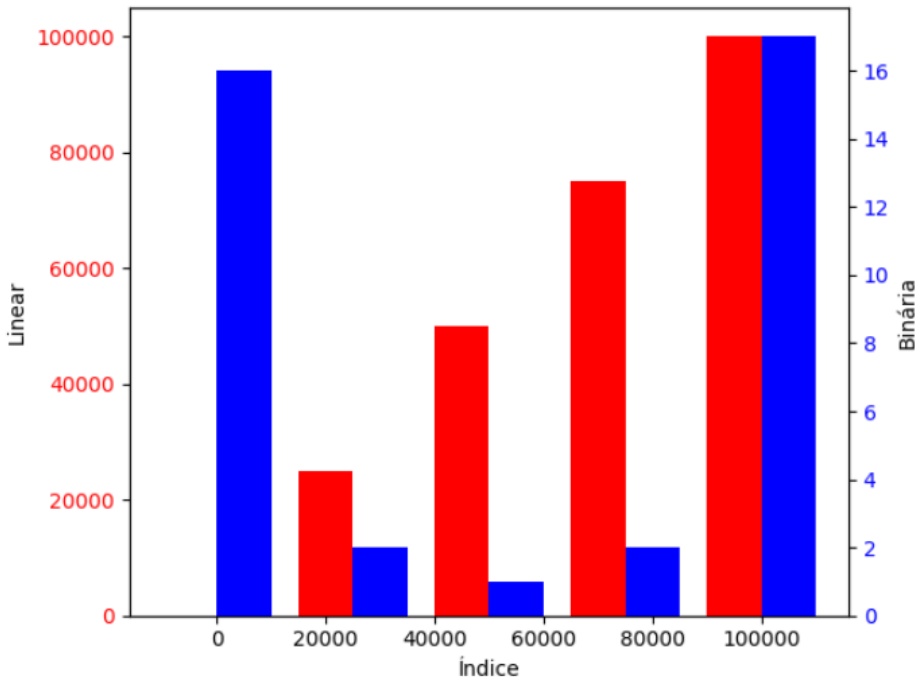


Gráfico de índice consultado × quantidade de iterações referente a cinco consultas distintas com algoritmo de busca linear (esquerda) e binário (direita). Produzido pelo autor.

Os índices consultados foram estrategicamente escolhidos de modo a explorar a relação entre a posição do alvo e a quantidade de iterações necessária para encontrá-lo (Tabela I). As posições inicial (0), média (49.999) e final (99.999) são fundamentais para

demonstrar a lógica crescente da busca linear e medial da busca binária. Além disso, o primeiro (24.999) e o terceiro (74.999) quartis foram utilizados para expandir a análise e comparar o comportamento dos algoritmos em diferentes seções de um problema.

Tabela 1 – Quantidade de iterações até alcançar o índice durante a execução de cada algoritmo de busca.

Índice	Linear	Binária
0	1	16
24.999	25.000	2
49.999	50.000	1
74.999	75.000	2
99.999	100.000	16

Fonte: Produzida pelo autor.

Quanto ao tempo de execução das buscas de paradigma clássico, o comportamento dos gráficos se repete, sendo crescente para a busca linear e com ponto mínimo central para a busca binária. Entretanto, deve-se considerar que o custo de ordenar o vetor é contabilizado para esta durante a execução a fim de nivelar a comparação a partir da mesma base dados. A ordenação é realizada pela função nativa *sort()* da linguagem Python, que utilizada o método Powersort (MUNRO; WILD, 2018). Isso resulta em um tempo de espera maior nos casos em que o índice da posição consultada é menor ou igual a 49.999 (Figura 6). Então, se o elemento procurado estiver na primeira metade do vetor, o custo adicional dos cálculos de ordenação da busca binária pode superar o benefício de reduzir o número de comparações. Em listas pequenas, uma busca linear simples frequentemente termina mais rápido.

Figura 6 – Comparação do tempo de execução entre os algoritmos de busca linear e binária.

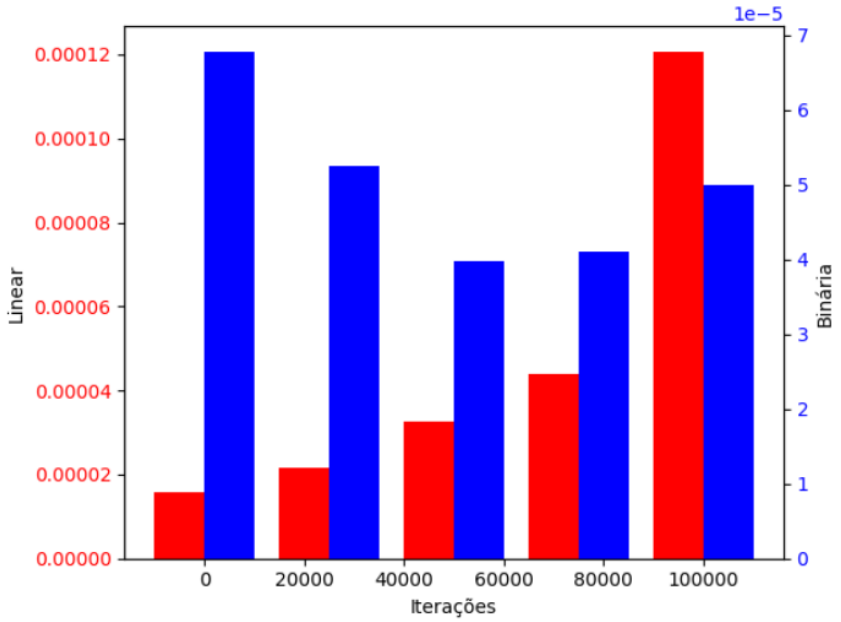


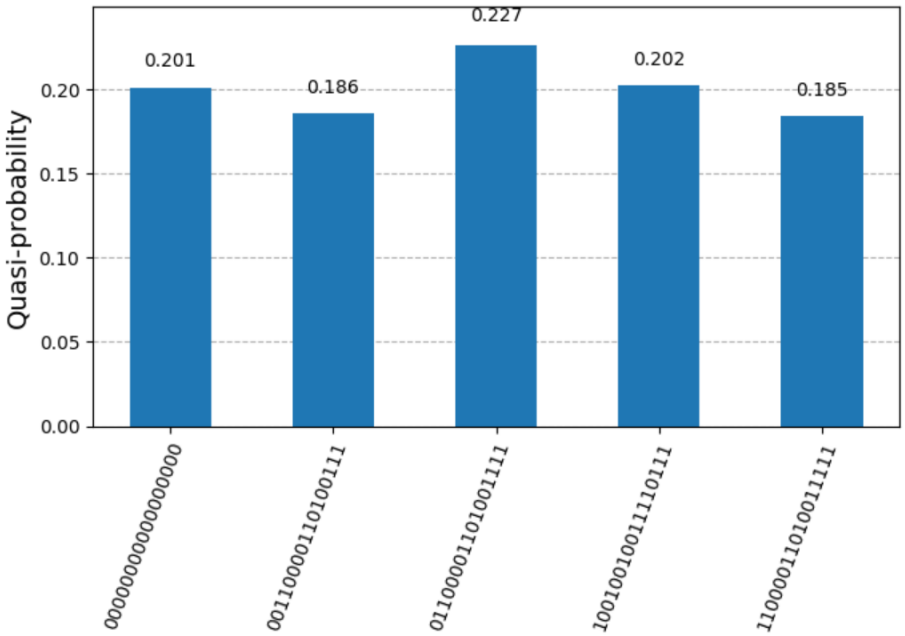
Gráfico de quantidade de iterações × tempo de execução em segundos referente a cinco consultas distintas com algoritmo de busca linear (esquerda) e binário (direita). Produzido pelo autor.

6.2 Algoritmo de Grover

Para a busca do paradigma quântico, os conceitos de posição e ordem não são aplicáveis, já que devido à superposição e o emaranhamento as probabilidades de cada estado são manipuladas simultaneamente. Sendo assim, múltiplos índices alvos, descritos em forma binária para representar os qubits envolvidos na execução, podem ser marcados durante uma busca e a medição final retornára o de maior probabilidade (Figura 7).

Múltiplos estados podem ser marcados no algoritmo de Grover porque o oráculo não está limitado a identificar apenas uma única solução e é projetado para aplicar uma inversão de fase a todos os estados que satisfazem uma determinada condição. Dessa forma, se existirem M soluções válidas entre os N estados possíveis, eles podem ser marcados simultaneamente e o processo de amplificação de amplitude do algoritmo aumenta a probabilidade de medir qualquer uma dessas soluções. Nesse cenário, a complexidade do algoritmo passa a ser aproximadamente $O(\sqrt{\frac{N}{M}})$ o que significa que a existência de mais soluções válidas pode inclusive reduzir o número de iterações necessárias. Essa característica é particularmente útil em situações nas quais o objetivo é encontrar qualquer elemento que satisfaça um critério e aplicada como ferramenta de decriptografia e de rede ao tratar redundância de nós.

Figura 7 – Execução do algortimo de Grover em que todas as posições de interesse são marcadas.



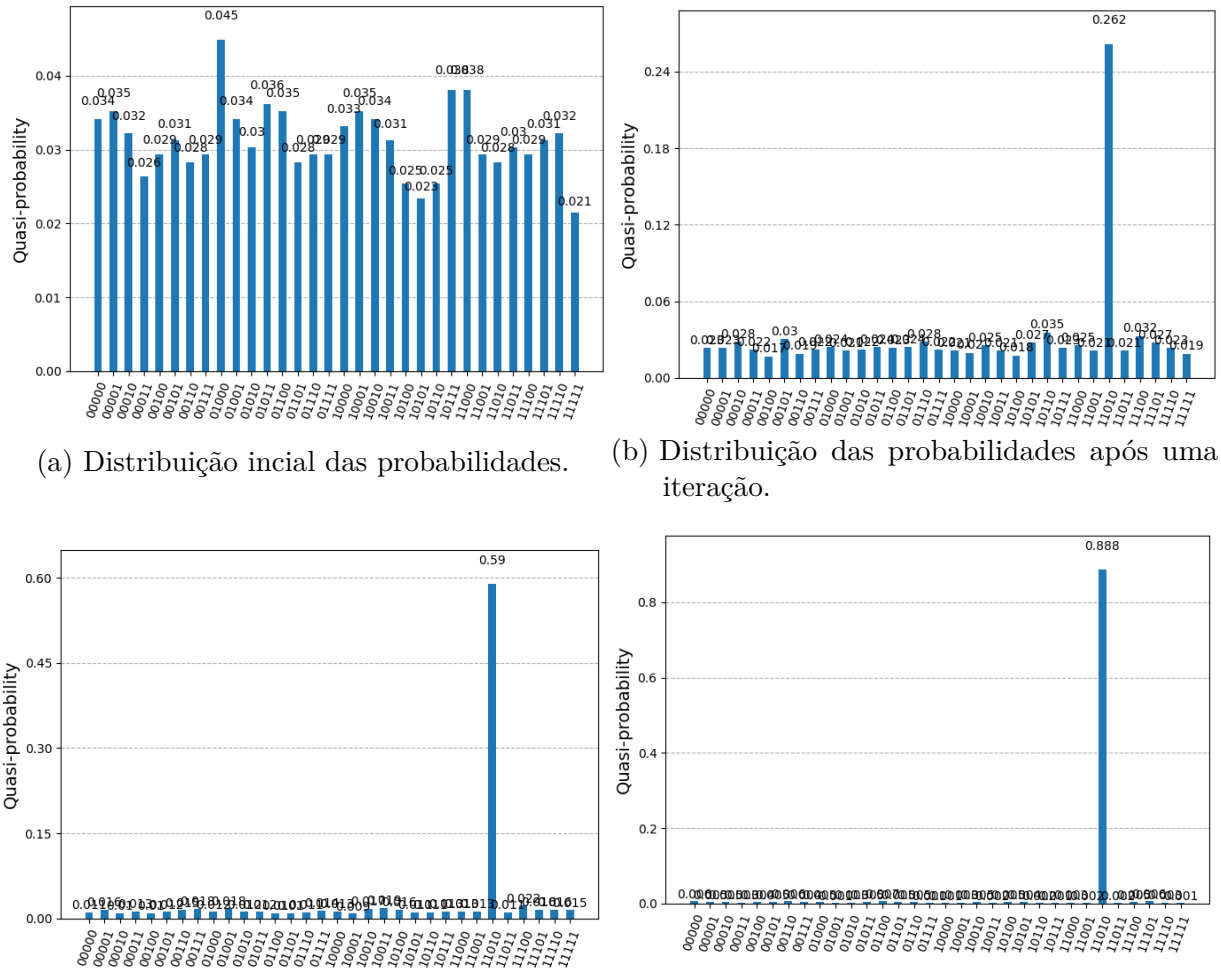
Distribuição de probabilidades após a execução do algoritmo de Grover em um problema com 17 qubits e 5 estados marcados. Produzido pelo autor.

Para representar 100.000 posições (chamadas de estados em problemas quânticos) são necessários 17 qubits ($2^{17} = 131.072$) e 127 ($N = \frac{\pi}{4}\sqrt{\frac{131.072}{5}}$) iterações do algoritmo de Grover para obter a probabilidade considerada ótima nos estados marcados. Neste ponto, o quadrado do módulo da amplitude dos demais estados devem ser iguais ou próximas a 0 de modo a serem desconsideradas na medição para o colapso e a soma de todos, incluindo os alvos, igual a 1.

6.2.1 Ruído

Com as ferramentas de visualização do Qiskit, é possível verificar o efeito de cada operação conjunta do oráculo e do difusor, que formam uma iteração do algoritmo de Grover. A figura 8 ilustra 4 passadas do algoritmo em um espaço com 32 estados representandos em 5 qubits no qual 11010 (26) foi marcado. Inicialmente as amplitudes de cada estado estão relativamente balanceadas, variando de 0,02 a 0,05, e a medição não colapsa para o alvo. Após apenas uma iteração, o componente associado ao estado marcado já se destaca notoriamente do conjunto e as demais tendem a uma uniformização abaixo desse valor. A simulação local desconsidera ruídos e devido a isso o comportamento do circuito é determinístico e linear, atingindo valor ótimo de acurácia após uma quantidade de iterações na escala da raíz quadrada do número de estados possíveis ($\sqrt{32} = 5,65685$).

Figura 8 – Evolução do Algoritmo de Grover.



(a) Distribuição incial das probabilidades.

(b) Distribuição das probabilidades após uma iteração.

(c) Distribuição das probabilidades após duas iterações.

(d) Distribuição das probabilidades após três iterações.

Distribuições de probabilidades após 4 execuções do algoritmo de Grover com quantidade de iterações crescente. Produzido pelo autor.

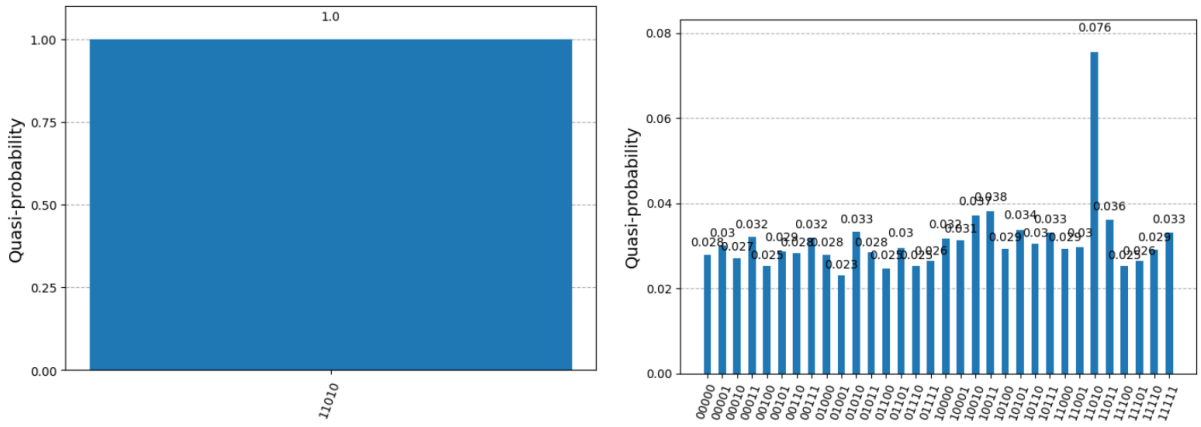
Enquanto as simulações do Algoritmo de Grover até então apresentadas desconsideram o ruído inerente ao circuito atrelado às portas quânticas, o processo de transpilação visa atenuar esse efeito na execução remota em máquinas reais. Assim, um circuito quântico escrito de forma abstrata pode ser transformado em um circuito equivalente executável

em um computador quântico real. Durante esse processo, o Qiskit adapta o circuito às restrições do hardware de destino, preservando seu comportamento lógico enquanto tenta otimizá-lo.

A transpilação envolve várias etapas, como a decomposição de portas quânticas complexas em um conjunto de portas básicas suportadas pelo dispositivo, o mapeamento dos qubits lógicos definidos pelo programador para os qubits físicos disponíveis no hardware e a otimização do circuito para reduzir o número total de portas e sua profundidade. Essas otimizações são importantes porque os computadores quânticos atuais são suscetíveis a ruídos e erros, de modo que circuitos menores tendem a produzir resultados mais confiáveis.

Em um cenário ideal, a solução converge para o alvo com 100% de precisão e o vetor de estado completo do sistema é representado apenas matematicamente para fins de validação teórica. Por meio da interface de comunicação com os serviços da IBM Quantum, é possível obter o resultado de uma execução real de programas quânticos com diferentes níveis de mapeamento de código para portas (Figura 9). O ganho de otimização tem um custo computacional elevado e níveis baixos, apesar de retornarem probabilidade menor de colapso do alvo, conseguem amplificar tal probabilidade suficientemente para que o colapso ocorra.

Figura 9 – Comparação do efeito do ruído e decoerência.



(a) Simulação local do Algoritmo de Grover.

(b) Execução do Algoritmo de Grover em backend real.

Fonte: Produzida pelo autor.

A execução de programas quânticos em backends reais por meio da biblioteca IBM Runtime é dividida em etapas chamadas de spans. Os básicos são a transpilação, o envio do trabalho para os servidores, espera em fila de execução, realização do circuito quântico, interpretação dos resultados e mitigação de erros. Os registros desses spans podem ser utilizados para analisar o desempenho do programa ao comparar o horário de início do primeiro span e de fim do último. No experimento acima, o tempo de execução do algoritmo de Grover em backend real foi de 4,08428 segundos, valor significantemente acima dos obtidos pelos algoritmos clássicos, mas que encapsula mais tarefas complexas.

7 DISCUSSÃO E LIMITAÇÕES DOS RESULTADOS

Os resultados obtidos confirmam as características teóricas esperadas de cada algoritmo. A busca linear apresentou crescimento proporcional ao tamanho da entrada,

demonstrando simplicidade de implementação, porém baixa eficiência para grandes conjuntos de dados. A busca binária evidenciou desempenho superior devido à sua complexidade logarítmica, embora dependa da ordenação prévia da estrutura de dados. Já o algoritmo de Grover demonstrou o potencial da computação quântica ao realizar buscas em espaços não estruturados com ganho quadrático, reduzindo a complexidade.

A escolha entre algoritmos de busca clássicos e quânticos depende não apenas de suas complexidades teóricas, mas também dos custos práticos envolvidos em sua execução. Quando executado em computadores clássicos por meio de simuladores, o algoritmo quântico incorre em custos adicionais significativos, incluindo a construção dos circuitos, a transpilação para o conjunto de portas suportado pelo hardware alvo e o tratamento dos resultados para serem interpretados em ambientes clássicos. Essas etapas frequentemente demandam mais recursos computacionais do que a própria busca, porém o valor dos algoritmos quânticos não se restringe à resolução direta desse problema.

Componentes como oráculos quânticos e operadores de difusão podem ser empregados como sub-rotinas em algoritmos híbridos e em problemas clássicos de otimização, verificação e amplificação de amplitude, permitindo explorar propriedades da computação quântica mesmo quando a solução completa não é inteiramente quântica. Dessa forma, para problemas de pequena ou média escala e em ambientes puramente clássicos, algoritmos tradicionais tendem a apresentar menores tempos de execução totais. Por outro lado, em espaços de busca muito grandes ou em aplicações que podem se beneficiar da integração de conceitos quânticos especializados, os algoritmos quânticos tornam-se alternativas promissoras, especialmente quando executados em hardware quântico capaz de explorar sua vantagem assintótica.

A simulação de algoritmos quânticos em hardware clássico é uma ferramenta essencial para pesquisa e desenvolvimento, porém apresenta limitações significativas decorrentes da própria natureza exponencial dos sistemas quânticos. A principal limitação está no consumo de memória: um sistema com n qubits é descrito por um vetor de estados contendo 2^n amplitudes complexas. Isso significa que, enquanto um computador clássico manipula um número de variáveis que cresce linearmente ou polinomialmente, a simulação exata de um circuito quântico exige recursos que crescem exponencialmente com o número de qubits.

8 CONSIDERAÇÕES FINAIS

Este trabalho apresentou uma análise comparativa entre algoritmos clássicos e quânticos de busca, utilizando como estudo de caso a busca linear, a busca binária e o algoritmo de Grover. A partir da implementação prática desses algoritmos em ambientes clássicos e quânticos, foi possível observar diferenças significativas entre os paradigmas computacionais, tanto do ponto de vista conceitual quanto em relação ao comportamento de desempenho.

Além da análise de desempenho, o trabalho possibilitou compreender conceitos fundamentais da computação quântica, como superposição, interferência, emaranhamento e amplificação de amplitude. A utilização do framework Qiskit permitiu explorar ferramentas modernas de desenvolvimento quântico, incluindo primitivas de execução, simuladores, serviços remotos e processos de otimização e redução dos efeitos de ruído. Também foi possível observar, por meio da comparação entre simulações ideais e execuções em hardware real, os impactos da decoerência e das limitações atuais dos dispositivos quânticos da era

NISQ.

Sob a perspectiva acadêmica, este estudo contribui ao apresentar uma abordagem introdutória e prática da computação quântica voltada para estudantes de computação, engenharia e matemática. Ao explicitamente relacionar algoritmos clássicos com seus análogos quânticos, o trabalho auxilia na compreensão da evolução dos paradigmas computacionais e aponta uma área em rápido crescimento científico e tecnológico.

Como trabalhos futuros, propõe-se a ampliação da análise para outros algoritmos quânticos e clássicos aplicados a problemas de otimização, grafos e criptografia, como o algortimo de fatoração de Shor (SHOR, 1997). Também é possível explorar execuções em computadores quânticos reais com maior quantidade de qubits, avaliando técnicas de mitigação de erro e estratégias mais avançadas de transpilação. Outra possibilidade consiste na investigação de algoritmos híbridos clássico-quânticos, combinando processamento clássico com rotinas quânticas especializadas, abordagem considerada promissora para os dispositivos quânticos contemporâneos. Por fim, futuras pesquisas podem aprofundar o uso didático de plataformas como a IBM Quantum Platform no ensino de computação quântica em cursos de graduação, contribuindo para ampliar o acesso a esse novo paradigma computacional.

REFERÊNCIAS

BULLYNCK, M. Histories of algorithms: Past, present and future. **Historia Mathematica**, v. 43, 2015. Citado na página 1.

CORMEN, T. H.; LEISERSON, C. E.; RIVEST, R. L.; STEIN, C. **Introduction to Algorithms**. 4th. ed. [S.l.]: Massachusetts Institute of Tecnology, 2009. Citado na página 1.

DAHL, O. J.; DIJKSTRA, E. W.; HOARE, C. A. R. **Structured programming**. [S.l.]: Academic Press, 1972. Citado na página 1.

FEYNMAN, R. P.; VERNON, J. F. L.; HELLWARTH, R. W. Geometrical representation of the schrödinger equation for solving maser problems. **Journal of Applied Physics**, v. 28, 1957. Citado na página 3.

FILATOV, S.; AUZINSH, M. Towards two bloch sphere representation of pure two-qubit states and unitaries. **National Library of Medicine**, v. 26, 2024. Citado na página 4.

GRIFFITHS, D. J.; SCHROETER, D. F. **Introduction to Quantum Mechanics**. 3th. ed. [S.l.]: Cambridge University Press, 2018. Citado na página 3.

GROVER, L. K. A fast quantum mechanical algorithm for database search. **Association for Computing Machinery**, v. 28, 1996. Citado 2 vezes nas páginas 1 e 5.

HENNESSY, J. L.; PATTERSON, D. A. **Computer Architecture: A Quantitative Approach**. 5th. ed. [S.l.]: Elsevier, 2012. Citado na página 1.

JESUS, G. F. de; SILVA, M. H. F. da; NETTO, T. G. D.; SOUZA, L. Q. G. F. G. de O.; CRUZ, C. Computação quântica: uma abordagem para a graduação usando o qiskit. **Revista Brasileira de Ensino de Física**, v. 43, 2021. Citado na página 8.

MUNRO, J. I.; WILD, S. Nearly-optimal mergesorts: Fast, practical sorting methods that optimally adapt to existing runs. **LIPIcs**, 2018. Citado na página 12.

NIELSEN, M. A.; CHUANG, I. L. **Quantum Computation and Quantum Information**. 10th. ed. [S.l.]: Cambridge University Press, 2010. Citado na página 2.

NISBET-JONES, P. B. R.; DILLEY, J.; HOLLECZEK, A.; BARTER, O.; KUHN, A. Photonic qubits, qutrits and ququads accurately prepared and delivered on demand. **New Journal of Physics**, v. 15, 2013. Citado na página 4.

QIU, D.; LUO, L.; XIAO, L. Distributed grover's algorithm. **Theoretical Computer Science**, v. 933, 2024. Citado na página 7.

RAY, P. Quantum simulation of dijkstra's algorithm. **International Journal of Advance Research in Computer Science and Management Studies**, v. 2, 2014. Citado na página 7.

SCHUMACHER, B. Quantum coding. **Physical Review A**, v. 51, 1995. Citado na página 2.

SHOR, P. W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. **SIAM Journal on Computing**, v. 26, 1997. Citado na página 17.

TOCCI, R. J.; WIDMER, N. S.; MOSS, G. L. **Sistemas Digitais: Princípios e Aplicações**. 11th. ed. [S.l.]: Pearson Education, 2011. Citado na página 4.

VENTURI, J. J. **Algebra Vetorial e Geometria Analítica**. 10th. ed. [S.l.]: Livrarias Curitiba, 2015. Citado na página 6.

WONG, T. G. **Introduction to Classical and Quantum Computing**. 1st. ed. [S.l.]: Rooted Grove, 2022. Citado 2 vezes nas páginas 1 e 3.

MINISTÉRIO DA EDUCAÇÃO
CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS
DEPARTAMENTO DE COMPUTAÇÃO - NG

FOLHA DE APROVAÇÃO DE TCC Nº 13 / 2026 - DECOM (11.56.03)

Nº do Protocolo: 23062.032284/2026-11

Belo Horizonte-MG, 23 de junho de 2026.

Guilherme Moreira de Carvalho

SIMULAÇÃO QUÂNTICA: Uma análise comparativa entre algoritmos clássicos e quânticos

Trabalho de Conclusão de Curso apresentado ao Curso de Engenharia de Computação do Centro Federal de Educação Tecnológica de Minas Gerais, do Campus Nova Gameleira, como parte do requisito para obtenção do grau de Engenheiro de Computação.

Aprovado em 12 de junho de 2026.

Banca Examinadora:

Prof. Dr. Tales Argolo Jesus
Centro Federal de Educação Tecnológica de Minas Gerais

Profa. Dra. Glívia Angélica Rodrigues Barbosa
Centro Federal de Educação Tecnológica de Minas Gerais

Prof. Dr. Bruno André Santos
Centro Federal de Educação Tecnológica de Minas Gerais

Belo Horizonte
2026

(Assinado digitalmente em 24/06/2026 15:04)
BRUNO ANDRE SANTOS
PROFESSOR DO MAGISTERIO SUPERIOR
DECOM (11.56.03)
Matrícula: 1459458

(Assinado digitalmente em 23/06/2026 11:43)
GLIVIA ANGELICA RODRIGUES BARBOSA
PROFESSOR ENS BASICO TECN TECNOLOGICO
DECOM (11.56.03)
Matrícula: 1058249

(Assinado digitalmente em 23/06/2026 09:42)
TALES ARGOLO JESUS
PROFESSOR ENS BASICO TECN TECNOLOGICO
DECOM (11.56.03)
Matrícula: 1663256

Visualize o documento original em **https://sig.cefetmg.br/public/documentos/index.jsp** informando seu número: **13**, ano: **2026**, tipo: **FOLHA DE APROVAÇÃO DE TCC**, data de emissão: **23/06/2026** e o código de verificação: **09b6f642c9**