



UNIVERSIDADE FEDERAL DO CEARÁ
CAMPUS QUIXADÁ
CURSO DE GRADUAÇÃO EM ENGENHARIA DE COMPUTAÇÃO

DAVID MACHADO COUTO BEZERRA

CIRCUITOS QUÂNTICOS VARIACIONAIS SIMÉTRICOS E OTIMIZAÇÃO
BAYESIANA PARA DETERMINAÇÃO DA ENERGIA DE ESTADO FUNDAMENTAL
NO TRIÓXIDO DE MOLIBDÊNIO

QUIXADÁ

2024

DAVID MACHADO COUTO BEZERRA

CIRCUITOS QUÂNTICOS VARIACIONAIS SIMÉTRICOS E OTIMIZAÇÃO BAYESIANA
PARA DETERMINAÇÃO DA ENERGIA DE ESTADO FUNDAMENTAL NO TRIÓXIDO DE
MOLIBDÉNIO

Trabalho de Conclusão de Curso apresentado ao
Curso de Graduação em Engenharia de Com-
putação do Campus Quixadá da Universidade
Federal do Ceará, como requisito parcial à
obtenção do grau de bacharel em Engenharia de
Computação.

Orientador: Prof. Dr. Antônio Joel Ra-
miro de Castro.

Coorientador: Prof. Dr. Thiago Werley
Bandeira da Silva .

QUIXADÁ

2024

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Sistema de Bibliotecas
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

B469c Bezerra, David Machado Couto.

Circuitos quânticos variacionais simétricos e otimização bayesiana para determinação da energia de estado fundamental no trióxido de molibdênio / David Machado Couto Bezerra. – 2024.
71 f.

Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Quixadá,
Curso de Engenharia de Computação, Quixadá, 2024.

Orientação: Prof. Dr. Antônio Joel Ramiro de Castro.

Coorientação: Prof. Dr. Thiago Werlley Bandeira da Silva.

1. Algoritmos. 2. Quântica. 3. Química. 4. Otimização. I. Título.

CDD 621.39

DAVID MACHADO COUTO BEZERRA

CIRCUITOS QUÂNTICOS VARIACIONAIS SIMÉTRICOS E OTIMIZAÇÃO BAYESIANA
PARA DETERMINAÇÃO DA ENERGIA DE ESTADO FUNDAMENTAL NO TRIÓXIDO DE
MOLIBDÊNIO

Trabalho de Conclusão de Curso apresentado ao
Curso de Graduação em Engenharia de Com-
putação do Campus Quixadá da Universidade
Federal do Ceará, como requisito parcial à
obtenção do grau de bacharel em Engenharia de
Computação.

Aprovada em: 16/09/2024

BANCA EXAMINADORA

Prof. Dr. Antônio Joel Ramiro de
Castro (Orientador)
Universidade Federal do Ceará (UFC)

Prof. Dr. Thiago Werlley Bandeira da Silva
(Coorientador)
Universidade Coorientador (SIGLA)

Prof. Dr. José Gadelha da Silva Filho
Universidade Estadual do Ceará (UECE))

À minha família, amigos e aos meus cachorros, Bela e Bob, por sempre estarem presentes e sendo fontes da minha energia e alegria, me apoiando em todas as minhas escolhas no caminho que estou seguindo.

AGRADECIMENTOS

À minha família, especialmente aos meus pais, Carlos Alberto e Miran Machado, aos meus avós, Maria do Socorro e Carlos Alberto, e à minha madrinha, Ciana Bezerra, por sempre me apoiarem nos momentos críticos e estarem presentes em toda a minha jornada nesta vida.

À minha cachorra e filha, Bela, por eu ter cuidado e criado durante 10 anos. Ela foi a razão de eu não desistir e continuar com um sorriso no rosto durante todo esse tempo.

Ao meu cachorro, Bob, por ser um grande e simpático bobalhão que sempre consegue arrancar meus sorrisos, e por sua disciplina e companheirismo.

Ao meu irmão, Lucas Machado, por ser meu exemplo desde pequeno, por ter cuidado de mim quando precisei, por me orientar e mostrar o que é certo ou errado, por ter compartilhado momentos assistindo filmes, jogando e, principalmente, por ter me feito rir desde a infância.

Aos meus primos, que são quase como irmãos: Felipe Machado, João Victor Couto e Valter Barros, que estão presentes nas minhas mais antigas memórias.

Aos meus queridos amigos do coração, que sempre estão presentes para me ouvir e me fazer rir nas situações mais complicadas: Anderson Moura, Javel Queiroz, Antonio César, Felipe Pedrosa, Paulo Aragão, Letícia Pianco, Carlos Kubernat, Lara Pianco, Igor André e Sabrina Felício.

Aos amigos de Quixadá, que fizeram companhia nos momentos no RU, nas filas, nos laboratórios, nas aulas, nas caminhadas de volta da faculdade e nas trilhas.

A todos aqueles amigos que foram importantes em minha trajetória, mas que hoje não estão mais presentes. Um grande obrigado, pois vocês foram fundamentais para o meu crescimento como pessoa.

Aos meus orientadores, Joel Ramiro e Thiago Werlley, por serem mais do que orientadores. Foram amigos que guiaram o meu caminho, sempre dando conselhos e ótimas aulas.

Aos excelentes professores que tive na faculdade: Criston Souza, Elvis Stancanelli, Arthur Araruna, Cristiano Bacelar e Michela Mulas. Vocês foram exemplos dentro da sala de aula e alimentaram minha vontade de estudar.

À turma de Sistemas Inteligentes em Controle e Automação de Processos, que me recebeu calorosamente em um período de adaptação no campus do Pici, por sua simpatia e por tornar as aulas momentos de diversão.

A todos, minha eterna gratidão.

"A natureza não é clássica, droga, e se você quer fazer uma simulação da natureza, é melhor que ela seja quântica, e, poxa, é um problema maravilhoso porque não parece ser nada fácil"
(Richard Feynman.)

RESUMO

Este trabalho investiga a aplicação do algoritmo *Variational Quantum Eigensolver* (VQE) para a molécula de Trióxido de Molibdênio (MoO_3), em foco na integração eficaz de diferentes estratégias de otimização em simulações quânticas. O estudo propõe a utilização de um estado de referência, definido por um algoritmo genético e combinado em um circuito variacional simétrico otimizado para *hardware*. Além disso, utiliza-se a otimização bayesiana para determinar a energia do estado fundamental do Trióxido de Molibdênio (MoO_3). Para demonstrar a validade do modelo construído foi realizado diferentes *benchmarks* comparando com diferentes modelos de circuitos e algoritmos de otimização. O processo foi dividido em três etapas: simulação com ruído, sem ruído e em um ambiente real. Na simulação sem ruído, foram comparados diferentes circuitos e otimizadores com a abordagem proposta neste trabalho. Em seguida, na simulação com ruído, os dois melhores otimizadores da etapa anterior foram utilizados para uma nova comparação. Finalmente, a execução em um computador quântico real da International Business Machines Corporation (IBM), limitada a 10 minutos, foi realizada. Os resultados gerais demonstraram a superioridade da estratégia proposta, destacando-se pelo tempo de convergência, simplicidade do circuito montado e eficiência, mesmo em condições limitadas, sugerindo que, com mais tempo, a precisão poderia ser ainda maior. Este estudo contribui, para o avanço dos estudos em computação quântica aplicados à química quântica, mostrando que a utilização de circuitos variacionais simétricos em conjunto com o otimizador bayesiano são o caminho correto para o VQE.

Palavras-chave: computação Quântica; vqe; otimização bayesiana; circuito variacional; química quântica.

ABSTRACT

This work investigates the application of the Variational Quantum Eigensolver (VQE) algorithm for the Molybdenum Trioxide (MoO_3) molecule, focusing on the effective integration of different optimization strategies in quantum simulations. The study proposes the use of a reference state, defined by a genetic algorithm and combined in a symmetric variational circuit optimized for hardware. Additionally, Bayesian optimization is used to determine the ground state energy of MoO_3 . To demonstrate the validity of the constructed model, different benchmarks were carried out, comparing various circuit models and optimization algorithms. The process was divided into three stages: simulation without noise, simulation with noise, and execution in a real environment. In the noiseless simulation, different circuits and optimizers were compared with the approach proposed in this work. Then, in the noisy simulation, the two best optimizers from the previous stage were used for a new comparison. Finally, execution on a real IBM quantum computer, limited to 10 minutes, was performed. The overall results demonstrated the superiority of the proposed strategy, standing out for its convergence time, simplicity of the constructed circuit, and efficiency, even under limited conditions, suggesting that with more time, the accuracy could be even higher. This study contributes to the advancement of quantum computing applied to quantum chemistry, showing that the use of symmetric variational circuits combined with Bayesian optimization is the right path for VQE.

Keywords: quantum computing; vqe; bayesian optimization; variational circuit; quantum chemistry.

LISTA DE ILUSTRAÇÕES

Figura 1 – Visualização de uma esfera de Bloch	23
Figura 2 – Fluxograma de um projeto usando Qiskit	28
Figura 3 – Estrutura eletrônica composta de prótons e elétrons	31
Figura 4 – Transformação do estado inicial $ 0111\rangle$ pelo operador de criação a_0^\dagger seguida pelo operador de aniquilação a_2	33
Figura 5 – Funcionamento do VQE	36
Figura 6 – O espectro de energia do hamiltoniano \hat{H} é deslocado pelo fator ω no novo operador	41
Figura 7 – Resultados dos cálculos de FS-VQE para estados eletrônico excitados de H2 e LiH	42
Figura 8 – Resultados das execuções do algoritmos para diferentes tamanhos N de qubits	43
Figura 9 – Taxa de sucesso para circuitos com excesso de parametrização	44
Figura 10 – Construção geral de circuitos eficientes	45
Figura 11 – Construção geral de circuitos eficientes	46
Figura 12 – Fluxograma dos procedimentos metodológicos	48
Figura 13 – Otimização do algoritmo genético	59
Figura 14 – Circuito Variacional	60
Figura 15 – Convergência para diferentes circuitos variacionais	61
Figura 17 – Convergência para diferentes otimizadores	62
Figura 18 – Convergência dos otimizadores utilizando simulação com ruído	63
Figura 19 – Circuito variacional transpilado para rodar no <i>hardware</i> alvo <i>ibm_kyiv</i>	64
Figura 20 – Convergência do Bayesian no <i>hardware</i> <i>ibm_kyiv</i>	64

LISTA DE TABELAS

Tabela 1 – Tabela comparativa - <i>Variational Quantum Eigensolver</i>	47
Tabela 2 – Termos amostrados do Hamiltoniano da molécula MoO ₃	66

LISTA DE ALGORITMOS

Algoritmo 1	– Script em Python para obter o Hamiltoniano de uma molécula	52
Algoritmo 2	– Função para converter operadores Pauli do formato da PennyLane para o do Qiskit	53
Algoritmo 3	– Função de aptidão do individuo	54
Algoritmo 4	– Função de custo de um circuito variacional	54
Algoritmo 5	– Função do algoritmo genético	55
Algoritmo 6	– Porta de Excitação única	56
Algoritmo 7	– Função <code>vf_symmetric</code> para construção do ansatz variacional	57

LISTA DE ABREVIATURAS E SIGLAS

ADAM	Adaptive Moment Estimation
BFGS	Broyden–Fletcher–Goldfarb–Shanno
D-Wave	D-Wave Systems Inc.
H ₂	Hidrogênio Molecular
IBM	International Business Machines Corporation
LiH	Hidreto de Lítio
MoO ₃	Trióxido de Molibdênio
NatGrad	Natural Gradient
NISQ	Noisy Intermediate-Scale Quantum era
Qiskit	Quantum Information Science Kit
STO-3G	Slater-type orbital with 3 Gaussians
UCCSD	Unitary Coupled Cluster with Single and Double excitations
VQE	Variational Quantum Eigensolver

LISTA DE SÍMBOLOS

α	Coeficiente complexo na superposição de estados de qubits
β	Coeficiente complexo na superposição de estados de qubits
Ψ	Função de estado quântico
$ 0\rangle$	Estado base $ 0\rangle$ de um qubit
$ 1\rangle$	Estado base $ 1\rangle$ de um qubit
\hbar	Constante de Planck reduzida
H	Hamiltoniano do sistema quântico
λ	Autovalor do Hamiltoniano (energia)
\mathbf{x}	Vetor de variáveis de otimização
U	Operador unitário em um circuito quântico
X	Operador Pauli-X para manipulação de qubits
Y	Operador Pauli-Y para manipulação de qubits
Z	Operador Pauli-Z para manipulação de qubits
\otimes	Produto tensorial de qubits
θ	Ângulo de rotação nas portas quânticas
ϕ	Ângulo de fase nas portas quânticas
\mathcal{X}	Conjunto viável de otimização
P_i	Momento de uma partícula
T_i	Energia cinética de uma partícula
$V(x,t)$	Potencial que depende da posição x e do tempo t
E_n	Energia associada ao estado n
Z_A	Número de prótons no núcleo A
r_{ij}	Distância entre as partículas i e j
$f(\mathbf{x})$	Função objetivo em problemas de otimização
\mathbf{p}_i	Quantidade de movimento do elétron i
a_i^\dagger	Operador de criação de estados quânticos

a_i	Operador de aniquilação de estados quânticos
H_{mol}	Hamiltoniano de uma molécula
$\psi_n(x)$	Autofunção na solução da equação de Schrödinger
ϵ_0	Permissividade elétrica do vácuo
e	Carga fundamental do elétron
π	Constante pi
m_e	Massa do elétron
M_A	Massa do próton A

SUMÁRIO

1	INTRODUÇÃO	17
1.1	Objetivos	19
<i>1.1.1</i>	<i>Objetivo geral</i>	19
<i>1.1.2</i>	<i>Objetivos específicos</i>	19
1.2	Organização do Trabalho	19
2	FUNDAMENTAÇÃO TEÓRICA	21
2.1	Circuitos Quânticos	21
<i>2.1.1</i>	<i>Bits Quânticos</i>	21
<i>2.1.2</i>	<i>Portas Quânticas</i>	23
<i>2.1.2.1</i>	<i>Portas de Entrada Única</i>	24
<i>2.1.2.2</i>	<i>Portas Quânticas de Entradas Múltiplas</i>	26
<i>2.1.3</i>	<i>Medições de Estados Quânticos</i>	27
<i>2.1.4</i>	<i>Qiskit</i>	28
2.2	Química Quântica	29
<i>2.2.1</i>	<i>Equação de Schrödinger</i>	29
<i>2.2.2</i>	<i>Hamiltoniano Eletrônico</i>	30
<i>2.2.3</i>	<i>Segunda Quantização</i>	32
<i>2.2.4</i>	<i>Transformação Jordan-Wigner</i>	34
2.3	Variational Quantum Eigensolver	34
<i>2.3.1</i>	<i>Método Variacional</i>	35
<i>2.3.2</i>	<i>Ansatz</i>	36
<i>2.3.2.1</i>	<i>Ansatz Simétrico</i>	37
<i>2.3.3</i>	<i>Otimizadores</i>	38
<i>2.3.3.1</i>	<i>Algoritmos de Otimização</i>	39
<i>2.3.3.1.1</i>	<i>Otimização Bayesiana</i>	40
3	TRABALHOS RELACIONADOS	41
3.1	<i>Folded Spectrum VQE: A Quantum Computing Method for the Calculation of Molecular Excited States (Tazi; Thom, 2024)</i>	41
3.2	<i>Avoiding local minima in variational quantum eigensolvers with the natural gradient optimizer (Wierichs et al., 2020)</i>	43

3.3	<i>Efficient symmetry-preserving state preparation circuits for the variational quantum eigensolver algorithm (Gard et al., 2020)</i>	44
3.4	Análise Comparativa	45
4	PROCEDIMENTOS METODOLÓGICOS	48
4.1	Revisão Bibliográfica	48
4.2	Discretização do Hamiltoniano	48
4.3	Construção do Circuito Variacional	49
4.4	Simulação do Algoritmo	49
4.5	Simulação de ruído e Mitigação no Circuito	50
4.6	Execução em Hardware Real	51
5	MÉTODOS E ALGORITMOS	52
5.1	Construção do Hamiltoniano	52
5.2	Algoritmo Genético para Estado Referencial	53
5.2.1	<i>Função de Aptidão</i>	53
5.2.2	<i>Algoritmo Genético</i>	55
5.3	Ansatz com Portas de Excitações Únicas	56
5.3.1	<i>Porta $A(\theta, \phi)$</i>	56
5.3.2	<i>Função $v_f_symmetric$</i>	57
6	RESULTADOS	58
6.1	Hamiltoniano do Trióxido de Molibdênio	58
6.2	Circuito Variacional Simétrico	58
6.3	Simulações	58
6.3.1	<i>Benchmark de Circuitos Variacionais</i>	59
6.3.2	<i>Benchmark de Otimizadores</i>	62
6.3.3	<i>Simulação com Ruído</i>	62
6.4	Execução em Hardware Quântico	63
7	CONCLUSÃO E TRABALHOS FUTUROS	67
7.1	Trabalhos Futuros	67
	REFERÊNCIAS	68

1 INTRODUÇÃO

A indústria de semicondutores se aproxima das limitações físicas, com transistores chegando à escala nanométrica, o que pode estagnar o crescimento exponencial do poder de processamento dos computadores observados no dia de hoje (Thompson; Spanuth, 2018). Para superar esses desafios, Richard Feynman propôs a ideia de um computador que utiliza a mecânica quântica em vez da teoria da informação clássica. Esse conceito, aprofundado por David Deutsch, demonstrou formalmente as vantagens dos computadores quânticos sobre os clássicos (Deutsch; Penrose, 1985). Essas inovações levaram pesquisadores a identificar vários problemas em que a computação quântica oferece uma vantagem significativa

Os computadores quânticos, como mencionado anteriormente, utilizam fenômenos da mecânica quântica para seu processamento. Os conceitos de superposição e entrelaçamento são centrais nessa abordagem. A superposição permite que qubits (unidades básicas de informação quântica) existam em múltiplos estados simultaneamente antes da medição, proporcionando um paralelismo intrínseco. Já o entrelaçamento, caracterizado pela correlação de partículas quânticas a distâncias arbitrárias, facilita a comunicação e transferência de estados quânticos. Estas propriedades são especialmente vantajosas para aplicações em inteligência artificial, redes neurais (Ramezani *et al.*, 2020) e no campo da química quântica (Cao *et al.*, 2019).

O estudo central da química quântica é a resolução da equação de Schrödinger, especialmente em sua forma independente do tempo, para sistemas moleculares (Cao *et al.*, 2019). Um desafio significativo nesse domínio é a manipulação e interpretação da função de onda, que aumenta em complexidade de forma exponencial à medida que o número de partículas cresce. Na computação clássica, diversas aproximações foram formuladas na tentativa de se aproximar da solução exata. Contudo, tais aproximações muitas vezes são limitadas, pois podem não contemplar a totalidade do espectro de estados possíveis. Atualmente, sistemas com mais de 30 elétrons apresentam enormes desafios (Blunt *et al.*, 2022). Tal limitação impacta setores como a indústria farmacêutica, que enfrentam obstáculos ao conduzir pesquisas mais profundas neste campo.

Esta nova corrida quântica pode ser comparada com a que se teve com relação à corrida espacial tendo agora como os principais competidores EUA e China, ambos investindo bilhões de dólares no potencial que essa tecnologia pode oferecer (Witt, 2022). As principais protagonistas dessa corrida têm sido empresas (Shankland, 2023), como a IBM, Google, D-Wave Systems Inc. (D-Wave), Amazon, etc. Um dos grandes motivadores dessa busca é o perigo

que um computador quântico pode oferecer ao sistema de segurança atual, com o algoritmo de Shor (SHOR, 1997), que, em teoria, poderia comprometer alguns sistemas de criptografia. Esses sistemas assumem certos fatores que são verdadeiros em computadores clássicos mas não em quânticos, como por exemplo que a fatoração de grandes números primos é uma tarefa muito difícil de se realizar.

A computação quântica enfrenta desafios significativos na implementação física dos qubits devido a perturbações não-lineares e instáveis, características da era *Noisy Intermediate Scale Quantum* (Noisy Intermediate-Scale Quantum era (NISQ)) (Lau *et al.*, 2022). Essas limitações dificultam a execução de algoritmos quânticos avançados, como a fatoração de números primos, que exigem correção de erros quânticos. Apesar disso, há debates sobre a confiabilidade dos processadores atuais em executar circuitos quânticos menos complexos que possam oferecer vantagens práticas. Nesse cenário, o *Variational Quantum Eigensolver* (VQE) (Peruzzo *et al.*, 2014) se destaca como uma solução promissora, utilizando circuitos quânticos curtos e algoritmos clássicos para simular sistemas quânticos complexos, por se capaz de calcular energia do estado fundamental de moléculas.

Embora o *Variational Quantum Eigensolver* (Variational Quantum Eigensolver (VQE)) tenha mostrado grande potencial na simulação de sistemas quânticos, a aplicação deste algoritmo em moléculas maiores e mais complexas ainda é limitada. A maioria dos estudos até agora tem se concentrado em sistemas relativamente simples, como por exemplo Hidreto de Lítio (LiH) e Hidrogênio Molecular (H_2) (Gard *et al.*, 2020), devido às restrições computacionais e à complexidade inerente do problema. No entanto, explorar o VQE em moléculas maiores, como o trióxido de molibdênio (MoO_3), exige o desenvolvimento de estratégias avançadas para lidar com o aumento da complexidade e o número de estados quânticos.

O trióxido de molibdênio (MoO_3) é um composto químico de grande interesse devido às suas propriedades catalíticas, ópticas e eletrônicas. Este óxido metálico é amplamente utilizado em diversas aplicações industriais, incluindo catalisadores para as áreas de catálise, eletroquímica e eletrônica, até óptica (Rocha *et al.*, 2024). A determinação precisa dos níveis de energia molecular de MoO_3 é essencial para entender e otimizar suas propriedades e aplicações.

Este trabalho tem como objetivo desenvolver o VQE focado no MoO_3 . A proposta inclui a implementação dos passos do algoritmo e a expansão do mesmo para obter outros estados de energia da molécula. O desenvolvimento de um circuito parametrizado que represente uma função de onda é crucial, pois captura os estados quânticos disponíveis. Além disso, a escolha de

um algoritmo otimizador eficiente para a busca dos estados de energia é crucial para o sucesso do projeto. Nesse contexto, o otimizador bayesiano mostrou-se particularmente adequado, devido à sua eficácia em lidar com sistemas de caixa preta e comportamentos não lineares. Por fim, uma análise crítica do processo será realizada para avaliar a precisão dos resultados obtidos e identificar possíveis melhorias na metodologia aplicada.

No escopo deste trabalho, foi realizada uma pesquisa inicial sobre a estrutura do trióxido de molibdênio (MoO_3) a ser utilizada. Em seguida, é preparado o Hamiltoniano da molécula com base nessa estrutura, otimizando-o para a aplicação desejada. Posteriormente, desenvolvemos um circuito simétrico parametrizado, investigando circuitos curtos e eficientes que representem a função de onda da molécula. Selecionando o otimizador bayesiano, realizamos simulações sem ruídos, comparando-o com diferentes algoritmos de otimização. Em seguida, os melhores resultados da etapa anterior são selecionados para uma simulação com ruído, baseada em um computador quântico real, a fim de demonstrar a resiliência da proposta e prepará-la para execução em *hardware* real.

1.1 Objetivos

1.1.1 *Objetivo geral*

Consequentemente, o objetivo geral do trabalho é implementar uma estratégia para o VQE, aplicada à molécula de MoO_3 , que o torne mais eficiente frente às restrições temporais e de recursos computacionais.

1.1.2 *Objetivos específicos*

- Desenvolver o Hamiltoniano do MoO_3 para aplicação no VQE.
- Implementar e otimizar um circuito quântico parametrizado para representar o espaço de Fock do MoO_3 .
- Realizar simulações e validar o desempenho do VQE em hardware quântico real.

1.2 Organização do Trabalho

Este trabalho é estruturado da seguinte maneira:

- Capítulo 2: apresenta a fundamentação teórica, elucidando conceitos relacionados à

computação quântica, química quântica e o VQE.

- Capítulo 3: discute trabalhos anteriores no domínio de algoritmos variacionais e suas aplicações, traçando paralelos e contrastes com a abordagem adotada neste trabalho.
- Capítulo 4: detalha a metodologia adotada para desenvolver a plataforma de emulação, incluindo design, implementação e avaliação. Também será apresentado o cronograma de atividades.
- Capítulo 5: detalha os algoritmos adotados para desenvolver o VQE otimizado para o MoO_3 .
- Capítulo 6: relata os resultados alcançados no trabalho, oferecendo uma análise das capacidades e limitações da estratégia adotada de emulação proposta.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta os conceitos fundamentais que embasam o trabalho proposto. Na Seção 2.1, é fornecida uma introdução aos circuitos de computadores quânticos, abordando qubits, portas e medições. A Seção 2.2 delinea uma breve exploração sobre química quântica e como pode ser trabalhado ela em um computador quântico. A Seção 2.3 discute o papel dos algoritmos quânticos variacionais, elucidando as etapas de sua utilização.

2.1 Circuitos Quânticos

Os circuitos quânticos são fundamentais na computação quântica, representando a maneira como os algoritmos quânticos são implementados. De maneira similar aos circuitos clássicos, que usam bits e portas lógicas como AND, OR e NOT para resolução de problemas, os circuitos quânticos operam com qubits e utilizam portas quânticas seguidas por uma medição final. Essas portas permitem a evolução dos estados ao longo do circuito, explorando superposição e entrelaçamento, características intrínsecas à mecânica quântica (Chuang; Yamamoto, 1995).

2.1.1 Bits Quânticos

Em computação quântica, a unidade fundamental de informação é o bit quântico, também conhecido como qubit. Ao contrário dos bits clássicos, que assumem valores de 0 ou 1, os qubits podem existir em uma combinação desses estados, representando assim um contínuo espectro de estados entre 0 e 1. Diferentes tecnologias têm sido exploradas para implementar qubits, sendo as seguintes as mais promissoras e utilizadas na indústria atualmente (Kurek, 2020):

- ***Superconducting Loops:*** Esta abordagem utiliza um circuitos supercondutores que possuem uma resistência mínima, onde uma corrente oscila. Pulsos de micro-ondas são utilizados para manipular os estados quânticos do loop.
- ***Trapped Ions:*** Nesta técnica, átomos individuais são presos e isolados em campos elétricos ou magnéticos. Lasers são então usados para manipular os estados quânticos dos átomos, aproveitando seus diferentes níveis de energia.
- ***Silicon Quantum Dots:*** Neste método, elétrons são confinados em pequenas regiões de silício, conhecidas como "pontos quânticos". Os estados quânticos destes elétrons são manipulados usando micro-ondas.

A informação contida em um qubit é representada por um vetor no espaço de Hilbert complexo. A notação de Dirac, introduzida pelo físico Paul Dirac, é comumente utilizada para representar esses vetores. Um estado genérico, Ψ , é denotado por $|\Psi\rangle$. Os estados base, $|0\rangle$ e $|1\rangle$, servem como equivalentes quânticos dos estados clássicos binários 0 e 1, e constituem a base padrão do espaço de Hilbert (Nielsen; Chuang, 2010). Qualquer estado neste espaço pode ser descrito como uma combinação linear dos estados base, conforme demonstrado na equação 2.1.

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle \quad (2.1)$$

Os coeficientes α e β são números complexos e precisam obedecer à relação de normalização:

$$|\alpha|^2 + |\beta|^2 = 1 \quad (2.2)$$

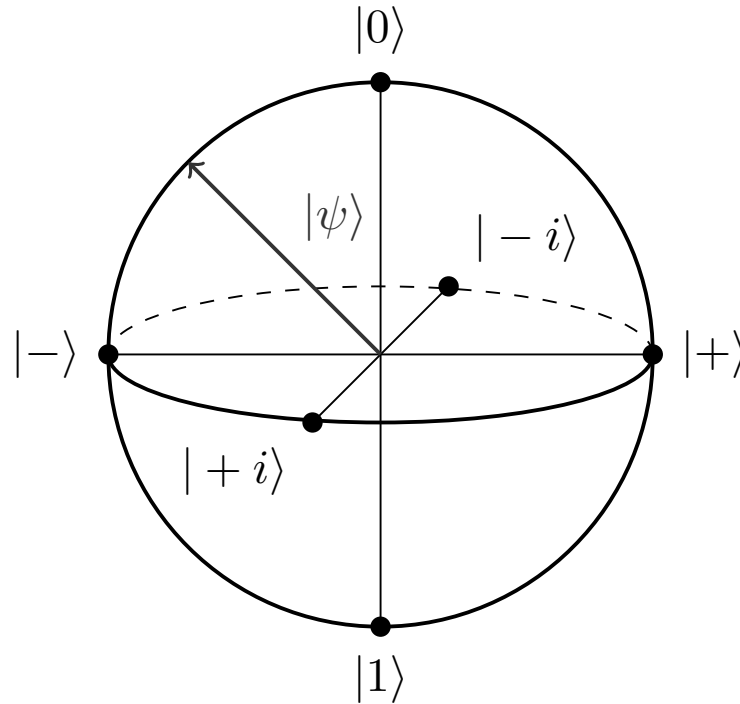
Essa relação assegura que o estado quântico esteja normalizado. Os estados que diferem de $|0\rangle$ e $|1\rangle$ estão em superposição dos estados base, uma propriedade fundamental que amplifica a capacidade dos algoritmos quânticos. Uma representação alternativa destes estados é através do seu vetor no espaço de Hilbert:

$$|\Psi\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \quad (2.3)$$

A esfera de Bloch proporciona uma representação visual intuitiva de um qubit. Neste modelo, qualquer estado quântico arbitrário de um qubit é ilustrado por um vetor unitário, que é posicionado sobre a superfície da esfera. Os estados bases $|0\rangle$ e $|1\rangle$ correspondem aos polos norte e sul da esfera, respectivamente. Assim, qualquer superposição desses estados é representada por um ponto na superfície da esfera, com suas coordenadas refletindo a combinação linear dos estados básicos. A Figura 1 apresenta um exemplo visual desta representação.

Essa representação gráfica facilita a compreensão dos estados quânticos, ilustrando como eles podem existir em qualquer ponto sobre a superfície da esfera, o que reflete as propriedades únicas de superposição e entrelaçamento dos qubits. Para sistemas compostos por múltiplos qubits, trabalha-se em um espaço de Hilbert de dimensão maior. A representação do vetor de estado se expande conforme a equação 2.4:

Figura 1 – Visualização de uma esfera de Bloch



Fonte: Elaborado pelo próprio autor

$$|\Psi\rangle = \bigotimes_{i=0}^n |\Psi_i\rangle \quad (2.4)$$

Onde \otimes denota o produto tensorial e n é o número de qubits no sistema. Esse modelo, que representa estados em espaços de Hilbert de dimensões maiores, não é a formulação geral para estados quânticos, pois no mundo real não se tem uma representação perfeita. Portanto, utiliza-se uma generalização de matrizes chamadas matrizes de densidade (Patel, 2023).

2.1.2 Portas Quânticas

As portas quânticas são os componentes fundamentais de um circuito quântico e permitem a realização de operações em qubits. A diferença entre as portas quânticas e as portas lógicas clássicas é que as portas quânticas operam em superposições de estados, o que permite o processamento quântico paralelo. Operações quânticas são governadas por operadores unitários (Nielsen; Chuang, 2010). A natureza unitária assegura a conservação da totalidade da probabilidade em sistemas quânticos. Matematicamente, um operador U é considerado unitário se, e somente se, o produto do operador e sua adjunta (conjugada transposta) é a matriz identidade:

$$UU^\dagger = U^\dagger U = I \quad (2.5)$$

Onde U^\dagger é a adjunta de U e I é a matriz identidade. As transformações quânticas, representadas por matrizes unitárias, garantem que os estados quânticos permaneçam normalizados. Na implementação física, as portas quânticas são realizadas de diversas formas, como lasers em sistemas de íons presos ou controle de correntes em circuitos supercondutores (Zagoskin, 2011). O objetivo principal é implementar a matriz unitária com precisão, preservando a integridade da informação quântica.

2.1.2.1 Portas de Entrada Única

As portas quânticas de entrada única atuam sobre um único qubit e são cruciais para a manipulação de estados individuais em um sistema quântico. Estas portas são as contrapartes quânticas das portas lógicas clássicas e são fundamentais para a execução de algoritmos em computadores quânticos. Dentre as portas de entrada única, destacam-se as Portas Pauli, nomeadas em homenagem ao físico Wolfgang Pauli. A Porta Pauli-X, comumente conhecida como Porta NOT quântica, realiza uma inversão de bit nos estados do qubit, trocando $|0\rangle$ com $|1\rangle$ e vice-versa (Aminian *et al.*, 2008). Matematicamente, é representada pela seguinte matriz e sua composição em um circuito:

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad (2.6)$$

No circuito quântico, esta porta é simbolizada por uma caixa marcada com um 'X', indicando a operação de inversão aplicada ao qubit, conforme demonstrado abaixo:

$$|q\rangle \text{ --- } \boxed{X} \text{ ---} \quad (2.7)$$

A Porta Pauli-Y é uma operação mais complexa, introduzindo uma rotação no espaço de Bloch que inclui uma fase imaginária. Ela é representada pela matriz:

$$Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \quad (2.8)$$

E no circuito quântico é representada como:

$$|q\rangle \text{ --- } \boxed{Y} \text{ ---} \quad (2.9)$$

Já a Porta Pauli-Z, ou Porta de fase, introduz uma fase de π ao estado $|1\rangle$ e é matematicamente descrita por:

$$Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad (2.10)$$

No diagrama de circuito correspondente, a porta é ilustrada da seguinte forma:

$$|q\rangle \text{ --- } \boxed{Z} \text{ ---} \quad (2.11)$$

Porém, dentre todas as portas de entrada única, a Porta Hadamard merece destaque especial. A capacidade de criar superposições de estados é um recurso essencial da computação quântica, e a Porta Hadamard permite esta transformação. Através dela, um qubit inicialmente em um estado base, como $|0\rangle$ ou $|1\rangle$, pode ser transformado em uma combinação linear destes estados, permitindo o processamento direto das amplitudes. Esta propriedade é fundamental para muitos algoritmos quânticos, como o Algoritmo de Grover e o Algoritmo de Shor. A matriz que representa a Porta Hadamard é:

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (2.12)$$

No diagrama de circuito correspondente, a porta é ilustrada da seguinte forma:

$$|q\rangle \text{ --- } \boxed{H} \text{ ---} \quad (2.13)$$

A aplicação desta porta resulta em estados que são cruciais para a exploração do "paralelismo" quântico e a obtenção de vantagens sobre a computação clássica. A compreensão profunda e a capacidade de manipular estas portas são, portanto, essenciais para qualquer avanço significativo na área de computação quântica.

2.1.2.2 Portas Quânticas de Entradas Múltiplas

No cenário da computação quântica, as portas que operam sobre múltiplos qubits desempenham um papel fundamental na manipulação conjunta e no controle de estados quânticos. Essas portas, diferentemente das de entrada única, são capazes de introduzir correlações quânticas entre qubits, fenômeno conhecido como emaranhamento. Tomando a Porta CNOT como exemplar dessa categoria, ela opera em dois qubits: um qubit controlador e um qubit alvo (Khalid *et al.*, 2004). A operação é condicional, de modo que a porta aplica uma transformação (no caso, uma operação NOT ou Pauli-X) no qubit alvo apenas quando o qubit controlador está no estado $|1\rangle$. Matematicamente, a Porta CNOT é representada por:

$$CNOT = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (2.14)$$

No circuito quântico, a representação da porta CNOT é dada por:

$$\begin{array}{ccc} |c\rangle & \text{---} \bullet \text{---} & |c\rangle \\ & | & \\ |t\rangle & \text{---} \oplus \text{---} & |t \oplus c\rangle \end{array} \quad (2.15)$$

Onde $|c\rangle$ é o qubit de controle e $|t\rangle$ é o qubit alvo. A saída do qubit de controle permanece inalterada, enquanto o estado do qubit alvo é invertido condicionalmente, baseando-se no estado do qubit de controle, representado pela operação XOR.

Essa capacidade de condicionar a operação de um qubit ao estado de outro é vital para a construção de lógicas complexas e algoritmos quânticos. A Porta CNOT, especificamente, é um recurso quintessencial para gerar estados emaranhados e realizar operações controladas em sistemas quânticos. O emaranhamento gerado por essa porta é a base para muitos protocolos quânticos avançados e algoritmos, reiterando a importância das portas quânticas de entradas múltiplas na arquitetura de um computador quântico.

2.1.3 Medições de Estados Quânticos

A computação quântica é intrinsecamente probabilística, diferenciando-se profundamente da computação clássica. Os algoritmos quânticos são probabilísticos, uma distribuição de probabilidade é produzida com base nas entradas do circuito e na distribuição de probabilidade induzida pelas portas (Haner *et al.*, 2016). Em contraste, na computação clássica, predominam os algoritmos determinísticos, nos quais se tem que o resultado já é a solução e não é uma preocupação. Nos circuitos quânticos, os algoritmos devem ser estruturados de modo que os resultados corretos possam ser obtidos com uma alta taxa de probabilidades, o que frequentemente exige que a computação seja repetida várias vezes antes de se conseguir medir os resultados com uma probabilidade suficientemente alta, como é feito em um dos algoritmos mais famosos de computação quântica, o algoritmo de busca de Grover (Grover, 1996).

A medição em sistemas quânticos é um ato irreversível, uma vez que a extração de informações de um sistema quântico destrói o próprio sistema que está sendo medido. Uma maneira de conceitualizar as medições quânticas é considerá-las como um conjunto de operadores (operadores irreversíveis) atuando no estado quântico, possibilitando a distinção do estado de cada qubit com uma determinada probabilidade (Nielsen; Chuang, 2010). Estes operadores são matrizes de projeção que mapeiam informações do domínio quântico para o domínio clássico. Após a medição, o qubit deixa de estar em superposição e é medido em um dos estados da base computacional. A escolha desses operadores e a complexidade do sistema quântico são determinantes para os erros de medição resultantes.

O diagrama a seguir ilustra a medição de um sistema com 5 qubits:

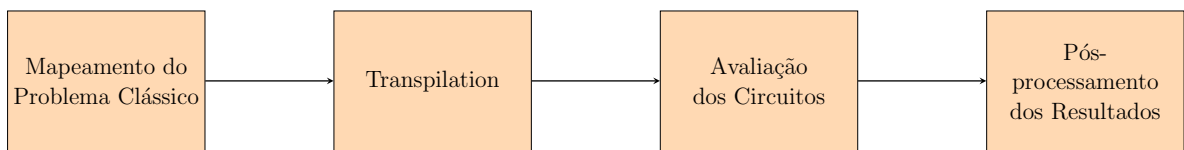


Esta representação indica que o qubit no estado $|\Psi\rangle$ é submetido a uma medição, resultando em um dos possíveis estados da base computacional, com a linha contínua convertendo-se em uma linha clássica após a medição.

2.1.4 Qiskit

Quantum Information Science Kit (Qiskit) é um kit de desenvolvimento de software de código aberto para computação quântica, criado pela IBM. Projetado para ser modular e extensível, o Qiskit permite que desenvolvedores e pesquisadores criem, simulem e executem algoritmos quânticos em computadores quânticos reais e simulados. A biblioteca está disponível para a linguagem *Python* e é composta por várias camadas, incluindo o Qiskit Aer, que permite a simulação de circuitos quânticos (Javadi-Abhari *et al.*, 2024). O fluxo típico de utilização do Qiskit segue um processo em quatro etapas, refletido em sua arquitetura de software mostrado na Figura 2:

Figura 2 – Fluxograma de um projeto usando Qiskit



Fonte: Elaborado pelo próprio autor

- **Mapeamento do Problema Clássico:** Um problema clássico é mapeado para um problema quântico, gerando circuitos que codificam o problema.
- **Transpilation:** Os circuitos gerados são transformados para serem adequados à execução no hardware alvo. Esse processo, conhecido como *transpilation*, envolve a reescrita dos circuitos sem compilá-los completamente em instruções clássicas do controlador.
- **Avaliação dos Circuitos:** Os circuitos transformados são então avaliados utilizando computações primitivas no backend escolhido, que pode ser um simulador ou um computador quântico real.
- **Pós-processamento dos Resultados:** Finalmente, os resultados obtidos são pós-processados para extrair uma solução para o problema original.

Esse fluxo pode ser iterativo, onde novos circuitos são gerados com base nos resultados obtidos de circuitos anteriores.

2.2 Química Quântica

Antes do advento da mecânica quântica, acreditava-se que os fenômenos físicos eram regidos exclusivamente pela teoria clássica, fundamentada nos princípios estabelecidos por Isaac Newton. De fato, observou-se que a mecânica clássica falhava em descrever comportamentos observados em escalas atômicas e subatômicas (Thakkar, 2021). Diferentemente dos sistemas macroscópicos, onde as leis newtonianas oferecem previsões precisas, as partículas em escala quântica, como elétrons e fótons, não seguem as trajetórias determinísticas previstas pela física clássica.

2.2.1 Equação de Schrödinger

A mecânica quântica tem como um de seus objetivos fundamentais resolver a equação de Schrödinger dependente do tempo, que é fundamental para descrever como o estado quântico de um sistema físico evolui ao longo do tempo:

$$\left(-\frac{\hbar^2}{2m} \frac{\partial^2}{\partial x^2} + V(x,t)\right) \Psi(x,t) = i\hbar \frac{\partial \Psi(x,t)}{\partial t} \quad (2.17)$$

Sendo:

- i é a unidade imaginária, onde $i^2 = -1$;
- \hbar é a constante de Planck reduzida, $h/2\pi$;
- m é a massa da partícula;
- $V(x,t)$ é o potencial que depende da posição x e do tempo t ;
- $\Psi(x,t)$ é a função de onda, que contém toda a informação sobre o sistema quântico.

A função de onda $\Psi(x,t)$ é de extrema importância na mecânica quântica, pois o seu módulo ao quadrado, $|\Psi(x,t)|^2$, fornece a densidade de probabilidade de encontrar a partícula na posição x no tempo t (Griffiths; Schroeter, 2018). A solução geral da equação de Schrödinger é dada por:

$$\Psi(x,t) = \sum_{n=1}^{\infty} c_n \psi_n(x) e^{-iE_n t/\hbar} \quad (2.18)$$

O conjunto de soluções individuais $\psi_n(x)$ forma uma combinação linear para formar a solução geral. Cada uma delas é chamada de autofunção, e a constante E_n é denominada autovalor, sendo a energia associada com uma solução n . Essa solução geral é a principal

característica da mecânica quântica, pois mostra que não se tem valores contínuos de energia, mas sim um conjunto quantizado de energias que podem assumir com uma probabilidade $|c_n|^2$, formando assim um espectro de energia. Uma maneira de se escrever a equação 2.17 para uma solução $\Psi_n(x)$ é usando a notação de Dirac:

$$\hat{H}|\Psi_n(x,t)\rangle = E_n|\Psi_n(x,t)\rangle \quad (2.19)$$

aonde

$$\hat{H} = \left(-\frac{\hbar^2}{2m} \frac{\partial^2}{\partial x^2} + V(x,t) \right) \quad (2.20)$$

Aqui, \hat{H} é o operador hamiltoniano que descreve a energia total do sistema, $|\Psi_n(x,t)\rangle$ é o estado quântico, e E_n é o autovalor de energia associado a esse estado. O hamiltoniano para uma partícula é representado na Equação 2.17, que inclui os operadores de momento e potencial. No entanto, para sistemas mais complexos, como moléculas, a equação de Schrödinger não pode ser resolvida analiticamente quando há mais de um elétron envolvido. Nessas situações, são necessárias aproximações, que serão discutidas na subseção 2.2.2.

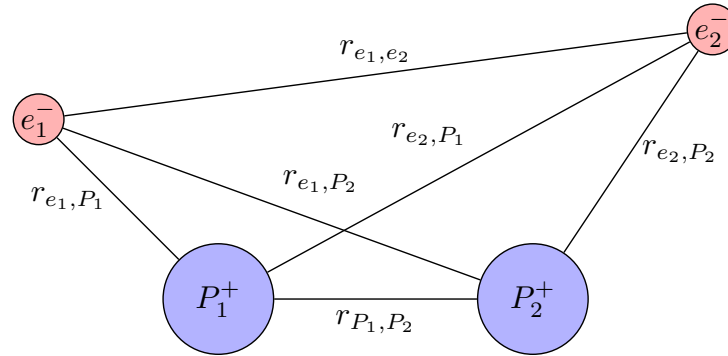
2.2.2 Hamiltoniano Eletrônico

Na química quântica, busca-se aplicar técnicas da mecânica quântica para analisar sistemas químicos, focando nas contribuições da estrutura eletrônica em moléculas e sólidos. Um dos objetivos é obter o estado de energia mínimo e o excitados de um hamiltoniano \hat{H}_{mol} relacionado a uma molécula, esse resultado permite utilizar para fazer previsões de reações químicas e propriedades espectroscópicas (Manenti; Motta, 2023). A Figura 3 demonstra a estrutura de eletrônica de uma molécula.

Para uma molécula qualquer o seu hamiltoniano pode ser representado pela soma dos termos cinéticos e potenciais de cada interação entre elétrons e núcleo da molécula, assim sendo a energia total do sistema, como mostrado na equação 2.21.

$$\hat{H}_{mol} = \hat{T}_e + \hat{T}_n + \hat{V}_{ne} + \hat{V}_{ee} + \hat{V}_{nn} \quad (2.21)$$

Figura 3 – Estrutura eletrônica composta de prótons e elétrons



Fonte: Elaborado pelo próprio autor

As letras e e n representam com relação aos elétrons ou núcleo da molécula. Os termos \hat{T} e \hat{V} representam o a energia cinética e potencial e são representados pelas equações 2.22 e 2.23.

$$\hat{T}_i = \frac{P_i^2}{2m} \quad (2.22)$$

Se tem que:

- P_i é a quantidade de movimento (momento) da partícula i ;
- m é a massa da partícula i .

$$\hat{V}_{ij} = \frac{q_i q_j}{4\pi\epsilon_0 r_{ij}} \quad (2.23)$$

Aonde:

- q_i é a carga da partícula i ;
- q_j é a carga da partícula j ;
- ϵ_0 é a permissividade do vácuo;
- r_{ij} é a diferença absoluta da distância entre os corpos i e j .

A forma expandida dos termos para uma molécula de N partículas negativas e M partículas positivas temos a seguinte forma expandida de 2.21:

$$\hat{H}_{\text{mol}} = \sum_{i=1}^N \frac{\mathbf{p}_i^2}{2m_e} + \sum_{A=1}^M \frac{\mathbf{P}_A^2}{2M_A} - \sum_{i=1}^N \sum_{A=1}^M \frac{e^2 Z_A}{4\pi\epsilon_0 r_{iA}} + \sum_{i=1}^N \sum_{j=i+1}^N \frac{e^2}{4\pi\epsilon_0 r_{ij}} + \sum_{A=1}^M \sum_{B=A+1}^M \frac{e^2 Z_A Z_B}{4\pi\epsilon_0 r_{AB}} \quad (2.24)$$

Onde:

- m_e é a massa do elétron;

- \mathbf{p}_i é a quantidade de movimento do elétron i ;
- M_A é a massa do próton A ;
- \mathbf{P}_A é a quantidade de movimento do próton A ;
- e é a carga fundamental;
- Z_A é o número de prótons no núcleo A .

A equação 2.24 pode ser simplificada conforme demonstrado por Born e Oppenheimer (1927), conhecida como a aproximação de Born-Oppenheimer. Nesta aproximação, assume-se que os núcleos, por serem muito mais pesados que os elétrons, têm movimento negligível. Assim, os termos de energia cinética dos núcleos são anulados ($\hat{T}_n = 0$) e o termo de potencial entre os núcleos (\hat{V}_{nn}) é tratado como uma constante, levando a uma nova equação simplificada.

$$\hat{H}_{\text{mol}} = \sum_i \frac{\mathbf{p}_i^2}{2m_e} - \sum_{i,A} \frac{e^2}{4\pi\epsilon_0 r_{iA}} + \sum_{i<j} \frac{e^2}{4\pi\epsilon_0 r_{ij}} \quad (2.25)$$

Uma conversão ainda pode ser feita com relação às unidades de medida. Em vez de trabalhar com o Sistema Internacional de Unidades (SI), utilizamos unidades atômicas (u.a.). Nessas unidades, a carga do elétron, a constante de Coulomb, e a massa reduzida do elétron são todas normalizadas a 1, simplificando significativamente a forma das equações (Manenti; Motta, 2023). O novo Hamiltoniano é dado por:

$$\hat{H} = - \sum_i \frac{\nabla_i^2}{2} - \sum_{i,A} \frac{Z_A}{r_{iA}} + \sum_{i<j} \frac{1}{r_{ij}}. \quad (2.26)$$

O Hamiltoniano desenvolvido até o momento é adequado para descrever sistemas moleculares, mas precisa ser convertido para uma forma que possa ser implementada em um computador quântico. Isso é feito por meio da segunda quantização, que reformula a Hamiltoniana em termos de operadores de criação e aniquilação.

2.2.3 Segunda Quantização

A segunda quantização visa simplificar a mecânica quântica de sistemas com grandes números de partículas, pois a complexidade dos cálculos cresce exponencialmente, tornando-os impraticáveis. Quando se trata de hamiltonianos eletrônicos, é mais adequado especificar em que estado cada elétron se encontra, ao invés de derivar a função de onda para cada elétron individualmente. Isso simplifica a descrição da função de onda eletrônica, tornando-a suficientemente

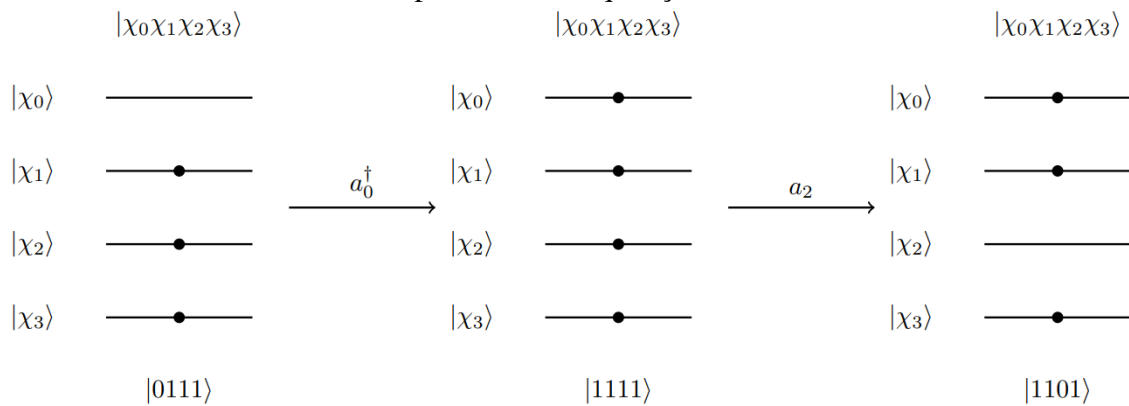
prática para ser representada por um registrador de um computador quântico (Cohen-Tannoudji *et al.*, 2019).

A função de onda é representada por uma *string* binária, onde o bit i -ésimo indica se um estado i está ocupado por um elétron (1) ou não (0). Como estamos lidando com férmions, não existe a possibilidade de dois elétrons estarem no mesmo estado. A formulação geral é escrita a seguir para um sistema de K partículas é:

$$|\mathbf{n}\rangle = |n_0, n_1, \dots, n_{K-1}\rangle \quad (2.27)$$

Com n_i sendo um valor binário representando a ocupação do estado i , os vetores utilizados na segunda quantização fazem parte do espaço de Fock, contido no espaço de Hilbert. A manipulação dos estados é realizada por meio dos operadores de criação (a_i^\dagger) e aniquilação (a_i), que adicionam ou removem um elétron do estado i , respectivamente. Esses operadores obedecem às relações de comutação antissimétricas, em conformidade com o princípio de exclusão de Pauli, eles recebem o nome de operadores fermiônicos. Um exemplo de seu funcionamento pode ser visto na Figura 4.

Figura 4 – Transformação do estado inicial $|0111\rangle$ pelo operador de criação a_0^\dagger seguida pelo operador de aniquilação a_2



Fonte: Elaborado pelo próprio autor

O hamiltoniano em 2.26 agora pode ser convertido para a representação utilizando o espaço de Fock e a segunda quantização:

$$\hat{H}_{\text{mol}} = \sum_{p,q} h_{pq} a_p^\dagger a_q + \frac{1}{2} \sum_{p,q,r,s} h_{pqrs} a_p^\dagger a_q^\dagger a_r a_s \quad (2.28)$$

O termo h_{pq} representa as projeções da energia cinética e das interações entre o elétron e o núcleo. Já h_{pqrs} denota as interações entre elétrons (Cao *et al.*, 2019). A forma expandida de ambos pode ser expressa conforme a seguir:

$$h_{pq} = \int dx \chi_p^*(x) \left(\frac{\nabla^2}{2} - \sum_{A=1}^M \frac{Z_A}{|\mathbf{R}_A - \mathbf{r}|} \right) \chi_q(x), \quad (2.29)$$

$$h_{pqrs} = \iint dx_1 dx_2 \frac{\chi_p^*(x_1) \chi_q^*(x_2) \chi_s(x_1) \chi_r(x_2)}{|\mathbf{r}_1 - \mathbf{r}_2|}. \quad (2.30)$$

Onde $\chi_i(x)$ é a função de onda que representa os orbitais atômicos de acordo com as coordenadas x para um certo orbital i . A formulação deste Hamiltoniano ainda precisa ser convertida para um computador quântico devido a problemas de localidade, exigindo assim uma conversão apropriada.

2.2.4 Transformação Jordan-Wigner

A solução para o problema de localidade é representar o Hamiltoniano como uma série de produto de tensores entre operadores de Pauli. No trabalho de Jordan e Wigner (1928), propõe-se um mapeamento conforme mostrado a seguir:

$$a_p^\dagger \xleftrightarrow{\text{JW}} |1\rangle \langle 0| \otimes Z_{p-1} \otimes \cdots \otimes Z_0, \quad (2.31)$$

$$a_p \xleftrightarrow{\text{JW}} |0\rangle \langle 1| \otimes Z_{p-1} \otimes \cdots \otimes Z_0, \quad (2.32)$$

A razão pela qual o operador Z aparece em outros qubits é devido ao fato de que os operadores fermiônicos causam uma mudança de fase quando aplicados. Com o mapeamento de Jordan-Wigner, se tem uma maneira de representar o Hamiltoniano em um computador quântico.

2.3 Variational Quantum Eigensolver

O Variational Quantum Eigensolver (VQE) é um dos algoritmos quânticos variacionais mais proeminentes e tem sido amplamente estudado devido ao seu potencial em química quântica e física de muitos corpos. O principal objetivo do VQE é determinar o valor de energia

do estado fundamental de uma molécula, aproveitando o poder dos sistemas quânticos e otimizadores clássicos (Peruzzo *et al.*, 2014). Com essa informação, é possível realizar cálculos mais precisos de reações químicas, que dependem da diferença de energia entre os estados iniciais e finais dos reagentes e produtos, determinar as estruturas e propriedades das moléculas.

A abordagem feita pelo algoritmo é híbrida, integrando componentes tanto da computação clássica quanto da quântica. Esta estratégia é escolhida devido à sua capacidade de representar estados quânticos de maneira mais precisa, aliada à eficácia dos métodos de otimização clássicos. A sinergia entre esses dois domínios facilita a identificação do estado fundamental do sistema, o objetivo central de qualquer algoritmo de otimização. Dentre as vantagens oferecidas por essa estratégia, destacam-se:

- i. O uso de um número reduzido de qubits.
- ii. A construção de circuitos quânticos com menor profundidade, implicando em menor número de portas.
- iii. Uma maior resiliência a ruídos, sejam eles coerentes ou incoerentes.
- iv. Uma representação mais precisa do estado fundamental.

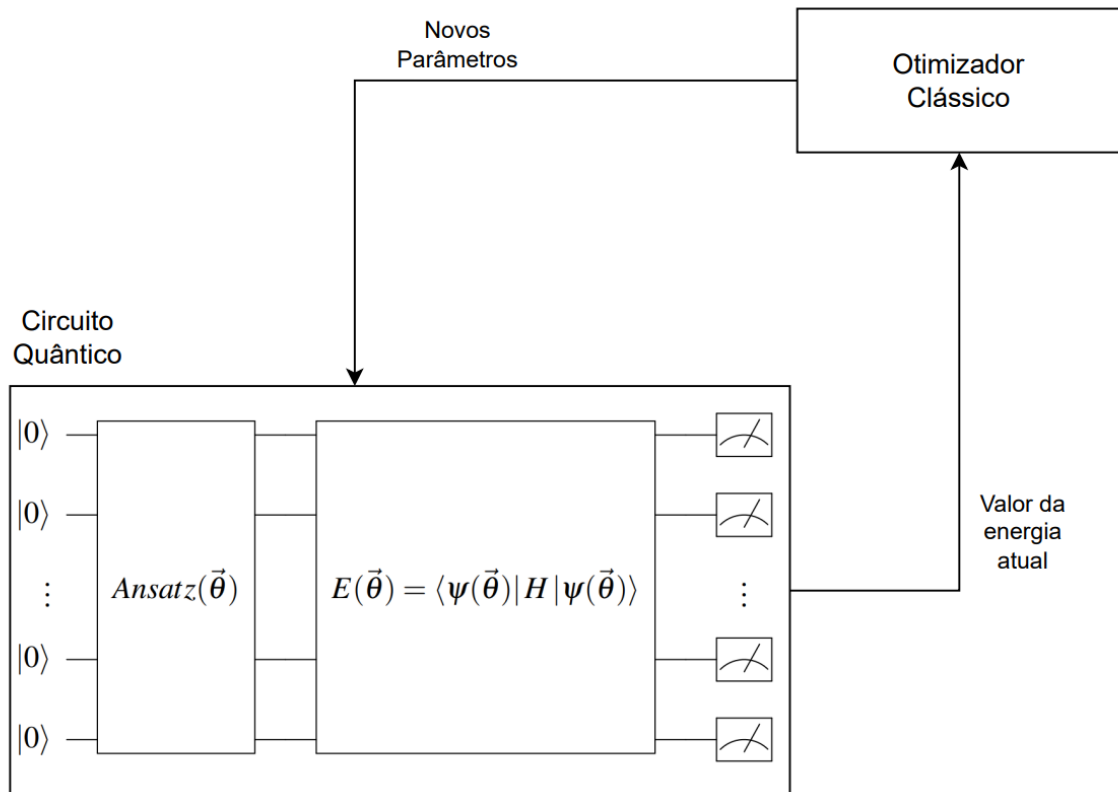
Esses benefícios são particularmente significativos para sistemas NISQ, onde as limitações técnicas ainda são uma barreira substancial.

O procedimento operacional do VQE começa com a preparação de um estado quântico parametrizado chamado de *Ansatz*, seguido pela definição de uma função de custo, que nesse caso vai ser o operador hamiltoniano. O valor obtido é então alimentado para um otimizador clássico que vai escolher novos parâmetros para o *Ansatz*. Este processo é repetido até que se obtenha a convergência para um valor mínimo desejado. O fluxograma do algoritmo pode ser visto na Figura 5.

2.3.1 Método Variacional

O método variacional é uma técnica poderosa utilizada no contexto do VQE para estimar o estado fundamental de um sistema quântico. A essência deste método reside na aplicação do Princípio Variacional, que garante que o valor esperado do Hamiltoniano, calculado para um estado quântico arbitrário ψ , fornece um limite superior para a energia do estado fundamental λ_0 do sistema (Manenti; Motta, 2023). Esse princípio pode ser formalmente expresso pela seguinte relação:

Figura 5 – Funcionamento do VQE



Fonte: Elaborado pelo próprio autor

$$\langle \psi | H | \psi \rangle = \sum_{i=1}^N \lambda_i |\langle \psi_i | \psi \rangle|^2 \geq \lambda_0 \quad (2.33)$$

Aqui, λ_i são os autovalores do Hamiltoniano H , e λ_0 representa o menor desses autovalores, correspondendo à energia do estado fundamental. O método variacional utiliza essa propriedade para aproximar λ_0 ao minimizar a energia esperada $\langle \psi | H | \psi \rangle$ sobre um conjunto de estados de teste parametrizados $|\psi(\mathbf{x})\rangle$. Assim, ao ajustar os parâmetros do estado ψ , o algoritmo VQE busca encontrar a configuração que minimize a energia, aproximando-se do valor da energia do estado fundamental (Peruzzo *et al.*, 2014).

2.3.2 Ansatz

A seleção de um *Ansatz* é crucial no projeto do algoritmo, pois define a abrangência do espaço de busca a ser explorado a cada interação. Ao escolher um *Ansatz*, considera-se fatores como velocidade de convergência, precisão do resultado e sensibilidade ao ruído. Existem três designs predominantes de *Ansatz* que se destacam:

- **Chemistry-inspired:** Projetados especificamente para a molécula em estudo, esses *Ansatz* incorporam conhecimentos químicos, focando em conFigurações eletrônicas específicas. Eles tendem a proporcionar maior precisão e rapidez, mas podem ser mais suscetíveis a ruídos.
- **Hardware-efficient:** Esses *Ansatz* são elaborados para otimizar a implementação em hardware quântico, utilizando portas que são tecnicamente mais simples de serem realizadas, independentemente das considerações químicas. Embora minimizem problemas relacionados a ruído, podem enfrentar desafios em termos de precisão e convergência.

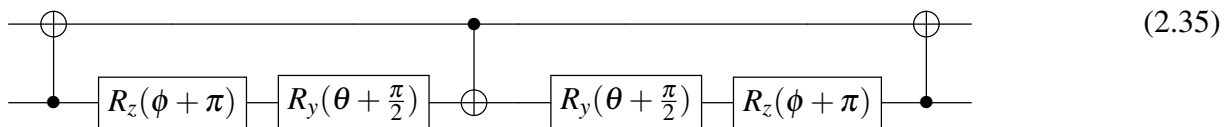
Cada *Ansatz* traz suas próprias vantagens e desafios, sendo a escolha guiada pelo contexto específico da aplicação e pelas características do hardware quântico utilizado (Fedorov *et al.*, 2021). A escolha deste trabalho foi por optar em um *ansatz* que fosse utilizar de uma abordagem híbrida.

2.3.2.1 Ansatz Simétrico

Em (Gard *et al.*, 2020), foi proposto um circuito quântico inspirado em química, utilizando rotações de Givens, o que permite a preservação de propriedades importantes como a conservação do número de partículas no sistema. Uma das portas centrais desse circuito é a porta quântica $A(\theta, \phi)$, que é descrita pela seguinte matriz:

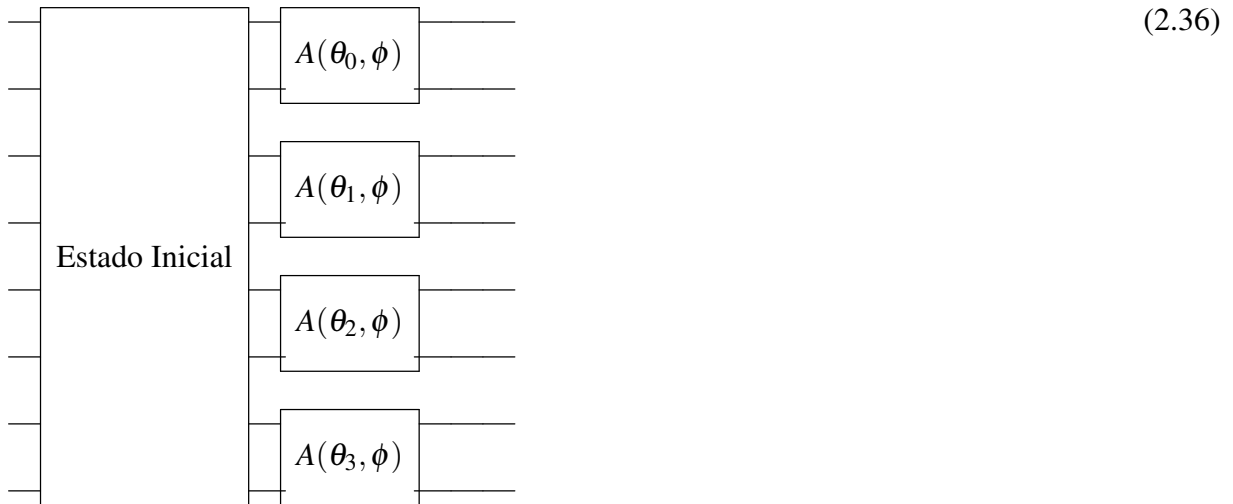
$$A(\theta, \phi) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \theta & e^{i\phi} \cdot \sin \theta & 0 \\ 0 & e^{-i\phi} \cdot \sin \theta & -\cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (2.34)$$

Esta matriz é projetada para preservar as simetrias de spin e a conservação do número de partículas no sistema, bem como o emaranhamento entre os orbitais. Ela pode ser implementada por meio do seguinte circuito quântico:



Este circuito estabelece um emaranhamento controlado entre os orbitais, criando um subconjunto específico dentro do espaço de Hilbert. Essa característica é essencial para uma

busca mais eficiente pelo estado de energia fundamental da molécula. O *ansatz* que foi adotado segue o seguinte formato:



O estado inicial do circuito é conFigurado por meio de uma *bitstring* que define quais portas X serão aplicadas com base no valor de cada bit: um valor de 1 resulta na aplicação de uma porta X no respectivo qubit. Este *ansatz* se destaca pela sua eficiência em termos de profundidade do circuito e número de parâmetros. Independentemente do número de qubits utilizados, a profundidade do circuito permanece constante, e o número de parâmetros necessários é limitado a $\frac{n}{2} + 1$, o que permite uma exploração eficaz do espaço de busca com menor custo computacional.

2.3.3 Otimizadores

Um problema de otimização clássica pode ser descrito como

$$\text{Min}_{\mathbf{x}} f(\mathbf{x}) : \mathbf{x} \in \mathcal{X} \quad (2.37)$$

onde o vetor $\mathbf{x} = (x_1, x_2, \dots, x_n)$ é a variável de otimização e f sendo a função objetivo (Boyd; Vandenberghe, 2004). Cada uma dessas variáveis, x_i , pode ser ajustada para minimizar a função objetivo $f(\mathbf{x})$ dentro do conjunto viável \mathcal{X} . Uma solução, é representada por \mathbf{x}^* , que é a variável que resulta no menor valor possível da função objetivo $f(\mathbf{x})$ entre todos os vetores que satisfazem as restrições impostas no problema.

A equação 2.37 pode ser considerada agora para o hamiltoniano da molécula levando em conta o método variacional:

$$\text{Min}_{\mathbf{x}} \langle \psi(\mathbf{x}) | H | \psi(\mathbf{x}) \rangle : \mathbf{x} \in \mathcal{X} \quad (2.38)$$

onde H é o Hamiltoniano do sistema, $|\psi(\mathbf{x})\rangle$ é o estado quântico parametrizado pela variável de otimização \mathbf{x} . Essa abordagem redefine o problema de otimização no contexto da computação quântica. O objetivo é encontrar os parâmetros \mathbf{x} que minimizam essa expectativa, o que corresponde a identificar o estado de menor energia do sistema, ou seja, o estado fundamental. Ao explorar diferentes estados quânticos possíveis dentro do conjunto viável \mathcal{X} , que é definido pelo *ansatz*, o algoritmo de otimização que vai realizar a busca é essencial, pois ele quem vai definir as escolhas de futuros estados até a convergência.

2.3.3.1 Algoritmos de Otimização

Os algoritmos de otimização desempenham o papel de escolha dos parâmetros ótimos em problemas de minimização, especialmente em algoritmos quânticos como o VQE. Esses algoritmos podem ser classificados em duas grandes categorias: baseados em gradiente (*gradient-based*) e sem gradiente (*gradient-free*).

Os métodos baseados em gradiente utilizam as derivadas da função objetivo em relação às variáveis de otimização para direcionar a busca pelo mínimo. Exemplos incluem:

- **L-BFGS-B**: Um algoritmo de otimização limitado por bordas que utiliza uma aproximação do gradiente para encontrar a solução ótima, sendo eficiente em problemas de alta dimensão (Byrd *et al.*, 1995).

No entanto, em sistemas ruidosos, comuns na computação quântica, o ruído pode introduzir instabilidades nas estimativas do gradiente, levando a direções de descida incorretas e possíveis convergências para mínimos locais indesejados ou até à divergência do algoritmo.

Por outro lado, os métodos sem gradiente não requerem o cálculo de derivadas e são mais robustos em ambientes ruidosos. Alguns exemplos incluem:

- **Nelder-Mead**: Um método de simplex que explora o espaço de soluções ajustando um conjunto de pontos para encontrar o mínimo (Nelder; Mead, 1965).
- **Powell**: Um método de direção conjugada eficiente para funções suaves e sem restrições (Powell, 1964).
- **COBYLA**: Um método de otimização sequencial sem derivadas que lida bem com restrições não lineares (Powell, 1994).

2.3.3.1.1 Otimização Bayesiana

Entre os diversos algoritmos de otimização, o bayesiano (Liu, 2023) se destaca como uma escolha ideal para situações em que se trabalha com uma função objetivo do tipo "caixa-preta". Esse tipo de função é caracterizado pela ausência de informações detalhadas sobre sua estrutura interna e gradientes, o que torna inviável a aplicação de métodos que dependam dessas informações. Além disso, o algoritmo bayesiano é particularmente eficaz em cenários onde a função objetivo é dispendiosa, seja por exigir um número limitado de avaliações ou por cada avaliação demandar um tempo significativo de execução. Outro aspecto relevante é a presença de ruído nas amostragens de f , uma condição frequentemente encontrada em experimentos de computação quântica. O processo de otimização bayesiana é composto por três passos principais, descritos de forma concisa a seguir:

1. **Modelo Probabilístico:** A partir das observações coletadas $(x_i, y_i = f(x_i))$ para $i = 1, \dots, t$, um modelo probabilístico é construído para estimar a função objetivo $f(x)$.
2. **Função de Aquisição:** O próximo ponto x_{t+1} é escolhido maximizando uma função de aquisição $u(x)$, que equilibra a exploração de novas áreas e a exploração de áreas promissoras. Neste trabalho, a função utilizada é a *Probabilidade de Melhoria* (PI):

$$PI(x) = P(f(x) \geq f(x_t^+) + \kappa) \quad (2.39)$$

onde $f(x_t^+)$ é o melhor valor observado até o momento e κ ajusta o equilíbrio entre exploração e exploração.

3. **Avaliação e Atualização:** O ponto x_{t+1} é avaliado para obter y_{t+1} , que é então utilizado para atualizar o modelo probabilístico. Este ciclo se repete até que um critério de parada seja atingido, como um número fixo de iterações ou a convergência para um nível de precisão desejado.

Essa abordagem permite uma busca eficiente e robusta, especialmente em cenários onde a função objetivo é custosa ou sujeita a ruído.

3 TRABALHOS RELACIONADOS

Neste capítulo, serão analisados diversos estudos da literatura que se concentram na aplicação de VQE na determinação de níveis de energia molecular. Esses estudos serão comparados com a proposta deste trabalho.

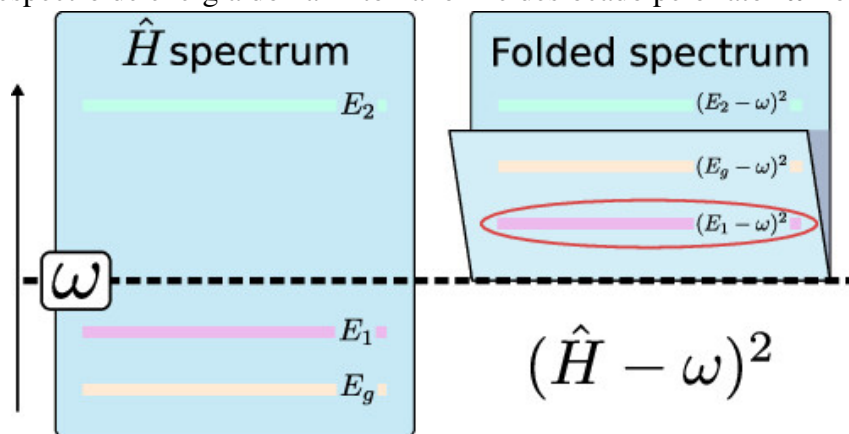
3.1 *Folded Spectrum VQE: A Quantum Computing Method for the Calculation of Molecular Excited States (Tazi; Thom, 2024)*

Em (Tazi; Thom, 2024) é proposto uma variação do VQE que realiza o cálculo de estados excitados para moléculas. A sua proposta se baseia de utilizar *Folded Spectrum* (FS) como uma extensão do VQE para o cálculo dos demais valores de energia, usando o mesmo circuito variacional. A eficiência do método é melhorada através do agrupamento de Pauli, que permite avaliar simultaneamente conjuntos de operadores de Pauli comutáveis, reduzindo o custo computacional.

No contexto do estudo apresentado, os autores detalham a implementação do algoritmo VQE adaptado ao método de *Folded Spectrum* (FS), nomeado agora para FS-VQE. Especificamente, eles modificam o operador Hamiltoniano tradicional substituindo \hat{H} pelo operador ajustado $(\hat{H} - \omega)^2$, onde ω é um deslocamento que ajusta o espectro de energia para focar em torno de um valor de energia específico, como mostrado na Figura 6. Essa modificação estratégica permite ao VQE direcionar o cálculo para estados excitados nas proximidades de ω , otimizando a eficiência do algoritmo em calcular energias perto deste ponto de referência.

Os autores do FS-VQE identificaram um desafio significativo de escalonamento

Figura 6 – O espectro de energia do hamiltoniano \hat{H} é deslocado pelo fator ω no novo operador

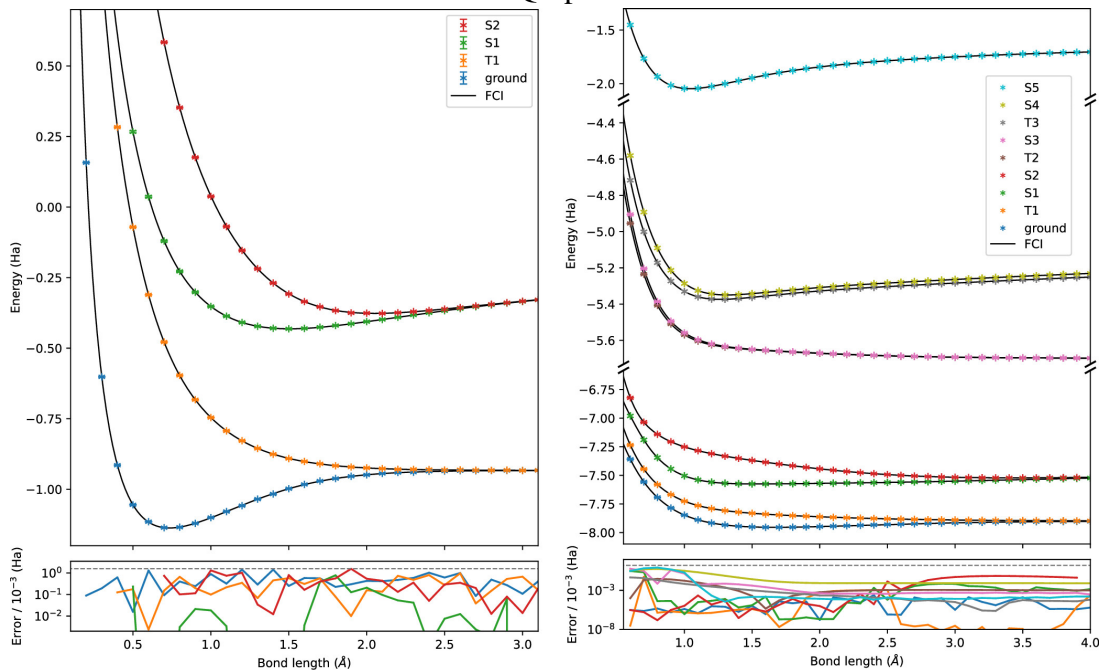


Fonte: (Tazi; Thom, 2024)

com o novo operador definido, onde o tamanho do operador escala quadraticamente, podendo alcançar uma complexidade de até $\mathcal{O}(N^8)$. Para mitigar este problema, eles propõem a utilização da Redução de Pauli, que emprega relações de anticomutação e comutação para reduzir a complexidade para $\mathcal{O}(N^6)$. Adicionalmente, o método de agrupamento de Pauli é empregado para dividir os operadores de Pauli em grupos que podem ser estimados simultaneamente em uma única medição, otimizando ainda mais o processo.

O estudo também empregou técnicas avançadas de robustez ao ruído e de mitigação de erros visando aprimorar a acurácia dos resultados. Utilizando a biblioteca open-source Qiskit, foram realizadas simulações tanto em ambientes ideais (sem ruído) quanto em condições realistas (com ruído), para modelar as interações moleculares. Foram utilizadas as moléculas de Hidrogênio (H_2) e Hidreto de Lítio (LiH) para modelar as interações moleculares. A aplicação dessas técnicas resultou em um sucesso relativo, como mostrado na Figura 7, evidenciando a viabilidade da abordagem para obter dados precisos, mesmo em face dos desafios impostos pelo ruído nos processadores quânticos.

Figura 7 – Resultados dos cálculos de FS-VQE para estados eletrônico excitados de H_2 e LiH



(a) Potential Energy Curves of H_2

(b) Potential Energy Curves of LiH

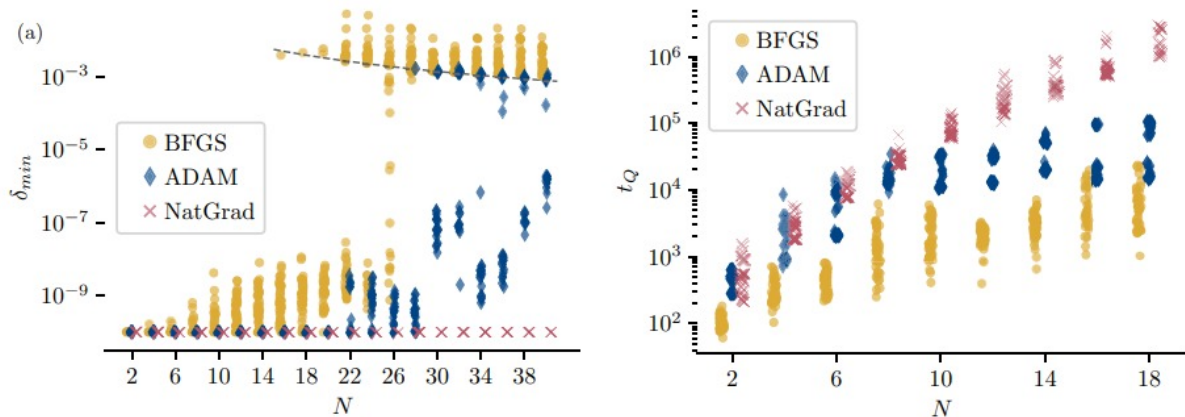
Fonte: (Tazi; Thom, 2024)

3.2 Avoiding local minima in variational quantum eigensolvers with the natural gradient optimizer (Wierichs *et al.*, 2020)

O artigo de (Wierichs *et al.*, 2020) realiza uma comparação detalhada entre diferentes algoritmos de otimização baseados em gradientes: Broyden–Fletcher–Goldfarb–Shanno (BFGS), Adaptive Moment Estimation (ADAM) e Natural Gradient (NatGrad). A performance desses algoritmos foi avaliada utilizando um *ansatz* do QAOA em modelos de Ising e XXZ, além de circuitos com excesso de parametrização capazes de quebrar a simetria do Hamiltoniano, em um ambiente de simulação.

O trabalho demonstrou a performance desses algoritmos para diferentes tamanhos de sistemas, variando de 2 até 40 qubits, comparando os otimizadores. O BFGS foi relatado como o otimizador com pior performance ao escalar, frequentemente ficando preso em mínimos locais. O ADAM apresentou boa convergência, porém, o número de *epochs* crescia rapidamente. O melhor desempenho foi observado com o NatGrad, que teve a maior taxa de sucesso e a menor quantidade de *epochs*, embora o custo por *epoch* fosse significativamente mais alto. O resultado pode ser visto na Figura 8, aonde N representa o número de qubits, δ é o error relativo e t_Q o tempo estimado de cada algoritmo.

Figura 8 – Resultados das execuções do algoritmos para diferentes tamanhos N de qubits

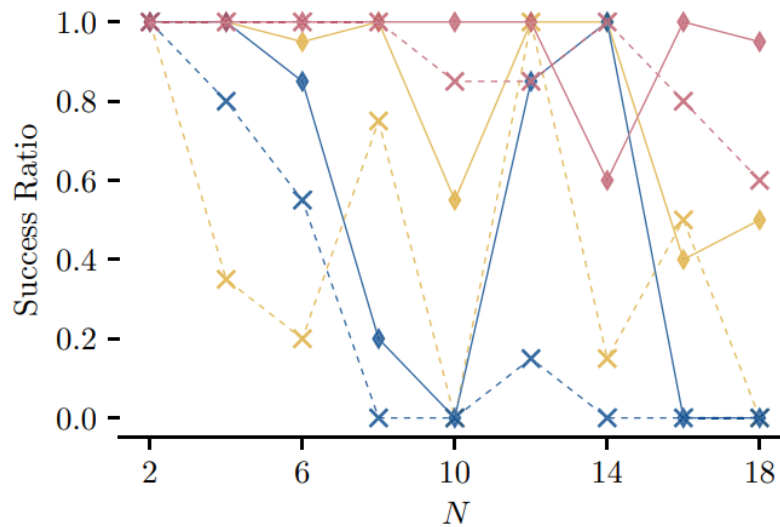


Fonte: (Wierichs *et al.*, 2020)

O segundo experimento analisou o impacto do excesso de parametrização no *ansatz* do VQE. O BFGS apresentou desempenho ruim, mesmo para sistemas pequenos, enquanto o ADAM também mostrou sinais de degradação à medida que o número de parâmetros aumentava. Por outro lado, o NatGrad demonstrou uma queda mínima de desempenho, destacando sua resiliência. Na Figura 9, é possível observar como a taxa de sucesso foi afetada pelo excesso de

parametrização.

Figura 9 – Taxa de sucesso para circuitos com excesso de parametrização



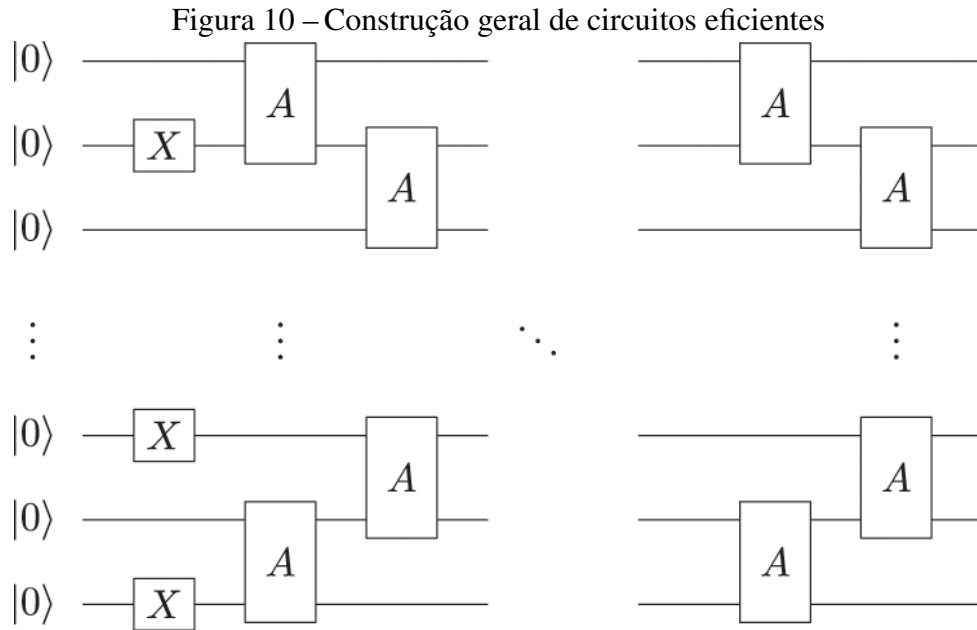
Fonte: (Wierichs *et al.*, 2020)

3.3 *Efficient symmetry-preserving state preparation circuits for the variational quantum eigensolver algorithm (Gard et al., 2020)*

(Gard *et al.*, 2020) propõem circuitos quânticos que preservam simetrias fundamentais, como número de partículas, spin total, projeção de spin e simetria de reversão temporal, durante a fase de preparação de estados do algoritmo VQE. A preservação dessas simetrias é crucial para garantir que os estados variacionais gerados permaneçam dentro do subespaço físico correto, resultando em soluções mais precisas e significativas. Além disso, os circuitos foram projetados para serem eficientes em termos de profundidade e contagem de portas, o que é vital para a implementação em dispositivos quânticos NISQ.

O circuito eficiente proposto para o caso geral de n orbitais e m férmions é composto por uma sequência de portas X e A , como mostrado na Figura 10. Primeiramente, aplicam-se portas X em m qubits para trazer o sistema para o subespaço correto de número de partículas. Em seguida, é realizada uma sequência de portas A para criar diferentes superposições. Através de experimentação, os autores descobriram que o número total de portas A necessárias para gerar todas as superposições possíveis é $\binom{n}{m}$. Como cada porta A contribui com dois parâmetros, o circuito final contém $2\binom{n}{m} - 2$ parâmetros, o número mínimo necessário para cobrir completamente o subespaço $H_{n,m}$.

Os resultados dos circuitos de preparação de estados propostos por (Gard *et al.*,



Fonte: (Gard *et al.*, 2020)

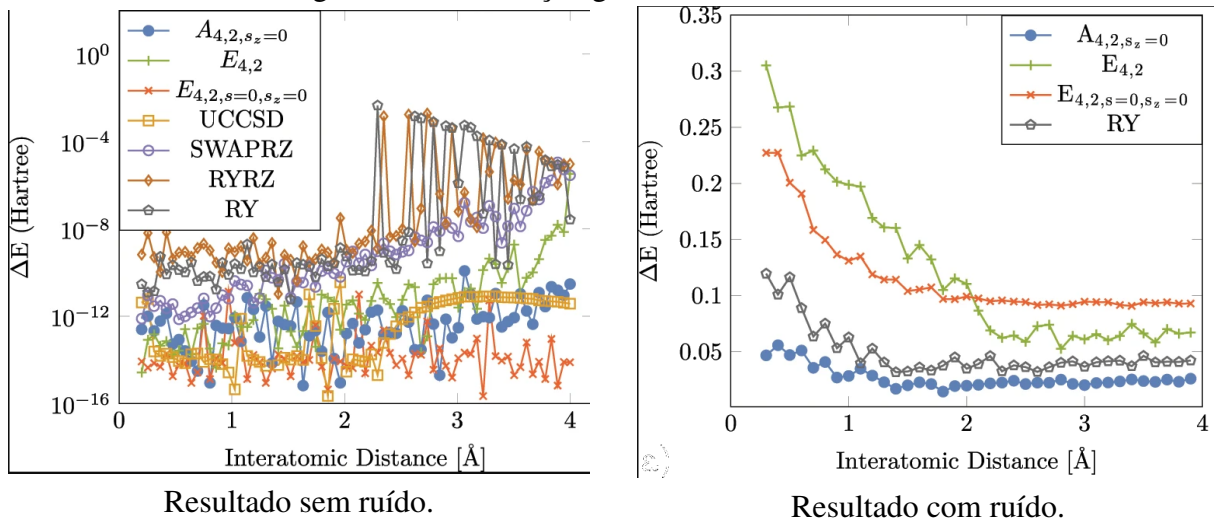
2020) mostraram-se altamente eficientes na simulação VQE para calcular a energia do estado fundamental da molécula de H_2 , utilizando a base Slater-type orbital with 3 Gaussians (STO-3G) e a transformação de Jordan-Wigner. Comparados aos ansätze tradicionais como RY, RYRZ e Unitary Coupled Cluster with Single and Double excitations (UCCSD), os circuitos propostos conseguiram atingir a energia do estado fundamental com alta precisão, utilizando o mínimo de parâmetros variacionais e um baixo número de portas CNOT. Em simulações com ruído, os circuitos também superaram o ansatz RY, mostrando que a preservação das simetrias de número de partículas e spin oferece uma vantagem significativa, mesmo em condições ruidosas. Os resultados, tanto para as simulações sem ruído quanto para aquelas com ruído, estão apresentados na Figura 11.

3.4 Análise Comparativa

No contexto de computadores quânticos para cálculos em química quântica, a literatura existente aborda várias estratégias e desafios associados à eficiência e escalabilidade. As questões associadas que foram abordadas em cada artigo podem ser usadas para desenvolver melhorias no algoritmo VQE.

Por outro lado, Tazi e Thom (2024) propõe uma nova técnicas para obter estados excitados. A pesquisa apresenta um enfoque técnico na otimização do circuito, hamiltoniano e da presença de ruídos e destaca a eficiência da sua abordagem, mas a técnica abordada precisa

Figura 11 – Construção geral de circuitos eficientes



Fonte: (Gard *et al.*, 2020)

de um conhecimento prévio dos estados excitados para utilizar do deslocamento do espectro.

Wierichs *et al.* (2020) foca na comparação técnica entres os algoritmos de otimização, tendo como *benchmark* a execução em um algoritmo variacional. Embora este ofereça uma visão sobre o desempenho sobre os algoritmos e qual tem a melhor performance, sendo o NatGrad, pode ser visto as desvantagens de utilizar um otimizador baseado em gradiente: sensível a ruído, custoso, a matriz relacionada ao NatGrad cresce em custo computacional polinomial.

Por outro lado, Gard *et al.* (2020) propõe um ansatz que, embora preserve simetrias fundamentais e seja eficiente para sistemas de pequeno porte, não é otimizado para moléculas maiores. O circuito proposto sofre com um crescimento exponencial no número de portas e parâmetros devido ao coeficiente binomial, o que limita sua aplicabilidade a sistemas de maior complexidade. Além disso, a abordagem não foi testada em moléculas mais complexas, o que limita sua eficácia em cenários mais exigentes.

Na proposta deste trabalho, busca-se abordar o VQE para o MoO_3 em computadores quânticos reais utilizando um circuito simétrico simplificado e o otimizador bayesiano. Essa abordagem visa demonstrar como trabalhar com o VQE em um ambiente real, priorizando a eficiência na preparação de estados e na otimização dos parâmetros, enquanto lida com as limitações impostas pelo hardware quântico disponível.

Na Tabela 1, é apresentado um resumo das características encontradas nos trabalhos relacionados e uma comparação com a proposta deste trabalho.

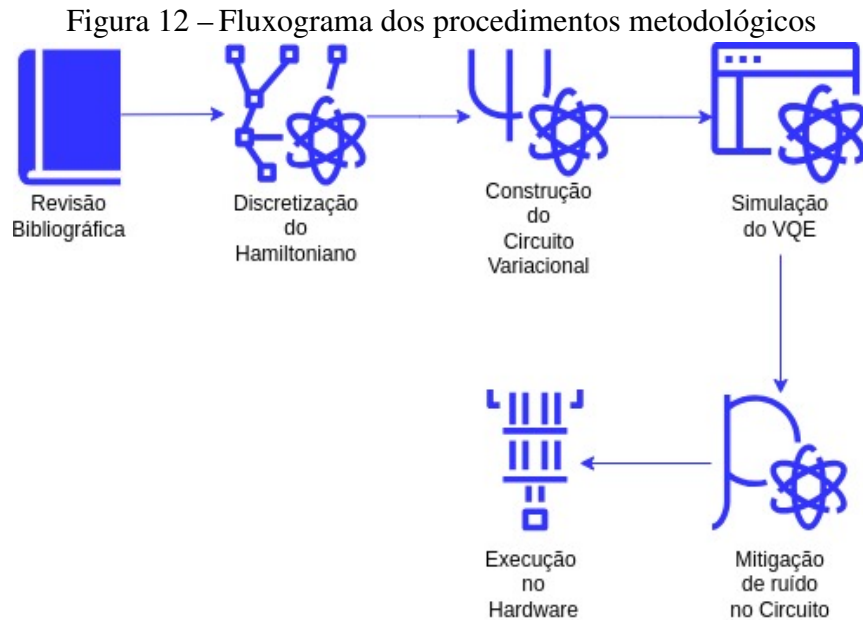
Tabela 1 – Tabela comparativa - *Variational Quantum Eigensolver*

Trabalho	Ansatz Otimizado	Mitigação de Ruído	Execução em Hardware Quântico	Comparação de Otimizadores	Comparação de Ansatz
Tazi e Thom (2024)	Parcialmente	Sim	Não	Não	Não
Waidyasooriya <i>et al.</i> (2022)	Não	Não	Não	Sim	Não
Gard <i>et al.</i> (2020)	Parcialmente	Não	Não	Não	Sim
Proposta deste trabalho	Sim	Sim	Sim	Sim	Sim

Fonte: Elaborado pelo próprio autor

4 PROCEDIMENTOS METODOLÓGICOS

Para atingir os objetivos delineados neste trabalho é essencial seguir uma série de etapas, conforme ilustrado no fluxograma da Figura 12.



Fonte: Elaborada pelo autor

4.1 Revisão Bibliográfica

A primeira etapa fundamental para o desenvolvimento deste trabalho foi realização de uma robusta revisão bibliográfica. Essa revisão foi focada em quatro áreas principais: os circuitos parametrizados (Ansatz) mais utilizados e suas vantagens, técnicas empregadas em cálculos de química quântica, otimizadores adequados para o algoritmo, e, por fim, estratégias de mitigação de ruído. A investigação desses temas na literatura científica proporcionou uma base sólida de conhecimentos, permitindo uma abordagem mais fundamentada e criteriosa ao longo de todo o projeto.

4.2 Discretização do Hamiltoniano

A discretização do Hamiltoniano é uma etapa essencial na implementação de algoritmos quânticos, como o VQE. Para essa tarefa, foram utilizadas as bibliotecas PennyLane e Qiskit, onde a PennyLane foi responsável pela construção inicial do Hamiltoniano molecular do MoO_3 , aproveitando suas capacidades de modelagem quântica. O Hamiltoniano foi gerado a partir da

estrutura molecular lida de um arquivo XYZ e determinou-se que a simulação requeria 20 qubits. Em seguida, os operadores quânticos e seus coeficientes foram processados e mapeados em operadores de Pauli ou identidades, garantindo a correta construção do produto tensorial.

O Hamiltoniano final foi representado como um objeto `SparsePauliOp` usando a `Qiskit`, o que facilitou sua manipulação e simulação nos circuitos quânticos. Esse Hamiltoniano, composto por 2239 termos de Pauli, assegurou uma representação precisa das propriedades quânticas do sistema molecular, enquanto otimizava o uso dos recursos computacionais disponíveis para a simulação.

4.3 Construção do Circuito Variacional

O circuito variacional foi construído com base no estado inicial determinado por um algoritmo genético e no *ansatz* desenvolvido especificamente para o VQE. O estado inicial foi preparado aplicando portas X em posições específicas dos qubits, conforme o *bitstring* de referência que foi gerado pela otimização do algoritmo genético no total de 10 gerações, garantindo a configuração correta dos estados eletrônicos iniciais. Esse estado inicial foi então combinado com um *ansatz* que utiliza portas de excitações únicas, implementadas pois representam corretamente o funcionamento de um sistema fermiônico.

O *ansatz* foi criado usando a função `vf_symmetric`, que organiza portas quânticas parametrizadas em camadas simétricas ao longo dos qubits. Essas portas foram implementadas de forma a garantir que cada camada possa representar excitações únicas, essenciais para capturar a dinâmica quântica do sistema molecular. O circuito resultante, composto pelo estado Hartree-Fock e pelo *ansatz*, foi então preparado para simulação, utilizando a função `compose` para integrar ambos de maneira eficiente. A construção desse circuito resultou em uma estrutura otimizada, com apenas 11 parâmetros e um tamanho constante. Esses aspectos são favoráveis tanto para a eficiência da otimização quanto para a redução dos efeitos de ruído durante a execução em hardware quântico.

4.4 Simulação do Algoritmo

A simulação sem ruído é uma etapa crucial para avaliar o desempenho de diferentes circuitos variacionais e otimizadores no contexto de algoritmos quânticos, como o VQE. Nessa fase, os circuitos variacionais são testados utilizando o estado de referência Hartree-Fock e o

ansatz desenvolvido. A simulação é realizada com uma variedade de otimizadores, incluindo L-BFGS-B, Nelder-Mead, POWELL, COBYLA e o otimizador bayesiano implementado pelo algoritmo `gp_minimize`. O critério de eliminação dos otimizadores é baseado na execução de 50 iterações e na obtenção de uma acurácia química de 0.0016 Hartrees. O principal objetivo é comparar a eficiência e precisão desses otimizadores, garantindo uma avaliação clara e objetiva das abordagens utilizadas.

O código implementado utiliza o estimador de vetores de estado da Qiskit para calcular a energia do sistema molecular para diferentes parâmetros do *ansatz*. A função `cf` é responsável por calcular a energia em cada iteração, e permite o uso de um *callback* para monitorar o progresso da otimização. Vários otimizadores são aplicados, com parâmetros iniciais definidos, para buscar a configuração de menor energia.

4.5 Simulação de ruído e Mitigação no Circuito

Após a simulação sem ruído, os melhores algoritmos que foram selecionados para a etapa seguinte, , que envolve a consideração dos efeitos do ruído. Nesta fase, uma nova simulação é realizada utilizando um *hardware* da IBM, especificamente o processador *Eagle r3* do *backend* `ibm_kyiv`, que possui 127 qubits, *Error per layered gate for a 100-qubit chain* (EPLG) de 1.8%, e uma capacidade de 5000 *Circuit Layer Operations Per Second* (CLOPS). Essa simulação incorpora os efeitos do ruído real nas operações quânticas, mapeando cada porta ideal para sua correspondente real no computador quântico. Esse mapeamento é essencial para refletir com precisão o comportamento do circuito quando executado em hardware quântico. As portas quânticas utilizadas no `ibm_kyiv` possuem os seguintes erros médios:

- **Echoed Cross-Resonance Gate (ECR):** 1.239e-2
- **Sqrt(X) Gate (SX):** 2.658e-4
- **X Gate:** 2.096e-4
- **Z Gate:** 0 (ideal)

Esses valores representam a taxa de erro associada a cada tipo de porta, permitindo que a simulação capture os efeitos do ruído com maior fidelidade, aproximando os resultados do que seria observado em uma execução real no hardware quântico. Os resultados dessa simulação servem como parâmetros críticos para a escolha final do otimizador, que serão executados no *hardware* quântico real. Além disso, estratégias de mitigação de ruído são escolhidas baseadas no resultado dessas simulação.

4.6 Execução em Hardware Real

Após a fase de simulações, onde os circuitos variacionais e otimizadores foram testados e otimizados tanto em ambientes sem ruído quanto com a consideração dos efeitos do ruído, chega-se à etapa final: a execução em hardware quântico real. Nesta etapa, o circuito variacional e o otimizador previamente selecionados são implementados em um dispositivo quântico disponibilizado pela IBM, respeitando uma restrição de tempo de execução de 10 minutos. O circuito é transpilado usando a biblioteca `qiskit.transpiler.preset_passmanagers` com o `generate_preset_pass_manager`, na qual permite traduzir as portas do circuito para o computador quântico alvo, nesse processo também é feita a formatação apropriada do hamiltoniano. A otimização escolhida foi a de primeiro nível sendo baseado em três fatores: quantidade de portas, profundidade do circuito e o quanto ele é equivalente ao circuito original.

A simulação com ruído evidenciou a necessidade de aplicar técnicas específicas de mitigação de erros para melhorar a precisão dos resultados. Dentre as opções consideradas, foi implementada a técnica de desacoplamento dinâmico, ativada com o uso de uma sequência $X_p X_m$ através da configuração `DynamicalDecouplingOptions(enable=True, sequence_type="XpXm")`. Além disso, foi utilizada a mitigação por extrapolação de ruído zero (ZNE), habilitada via `options.resilience.zne_mitigation = True`, proporcionando maior resiliência ao ruído inerente à simulação. Com a configuração definida, a execução foi iniciada como uma sessão real utilizando o comando `with Session(backend=backend) as session`, onde o `Estimator` foi instanciado com as opções previamente configuradas.

5 MÉTODOS E ALGORITMOS

Neste capítulo, será apresentado o desenvolvimento do ambiente de simulação. As seções a seguir descrevem as implementações e métodos utilizados para a simulação e análise dos resultados. Todo o código-fonte utilizado para a criação deste ambiente de simulação, bem como as etapas de processamento e análise, está disponível e pode ser acessado através do Google Colab, permitindo que outros pesquisadores repliquem e validem os experimentos descritos.

5.1 Construção do Hamiltoniano

Para obter o Hamiltoniano de uma molécula, é necessário utilizar um arquivo no formato `.xyz`, que fornece informações detalhadas sobre a estrutura geométrica da molécula. O algoritmo desenvolvido para essa finalidade está ilustrado no Algoritmo 1.

Algoritmo 1: Script em Python para obter o Hamiltoniano de uma molécula

```

1 def create_hamiltonian(xyz_path):
2     symbols, coordinates = qchem.read_structure(xyz_path)
3     H, NUM_QUBITS = qchem.molecular_hamiltonian(symbols,
4         coordinates)
5     hamiltonian_str = str(H)
6     hamiltonian = []
7     pattern = re.compile(r'\((( [-+ ]? \d* \. \d+ | \d+ ) \) \[([XYZI]) \d+ (? : \
8         s [XYZI] \d+ ) * \] ')
9
10    for line in hamiltonian_str.splitlines():
11        match = pattern.search(line)
12        if match:
13            constant = float(match.group(1))
14            raw_operator = match.group(2)
15            operator = convert_operator(raw_operator, NUM_QUBITS)
16            hamiltonian.append((operator, constant))
17
18    return qi.SparsePauliOp.from_list(hamiltonian), NUM_QUBITS

```

Fonte: Elaborado pelo autor.

O algoritmo apresentado converte a estrutura geométrica de uma molécula, fornecida em um arquivo `.xyz`, em um Hamiltoniano quântico utilizando a representação de Jordan-Wigner. A função `create_hamiltonian(xyz_path)` (linha 2) inicia o processo ao ler os símbolos atômicos e as coordenadas cartesianas da molécula, utilizando o método `qchem.read_structure(xyz_path)` (linha 3). Na linha 4, a função é empregada para calcular o Hamiltoniano molecular nomeada de `qchem.molecular_hamiltonian`, que é então

convertido para uma string (`hamiltonianz_str`, linha 5). A partir da linha 8, um padrão regex é utilizado para extrair os coeficientes e operadores do Hamiltoniano, que são posteriormente convertidos no formato padrão do Qiskit, conforme indicado nas linhas 9 a 14. Finalmente, a função retorna o Hamiltoniano no formato `SparsePauliOp` do Qiskit (linha 16).

A extração dos operadores Pauli é realizada utilizando uma expressão regular (*regex*) definida na linha 15 do algoritmo 1. Este *regex* é usada para identificar linhas na string do Hamiltoniano que contêm um termo de constante (o coeficiente de cada operador Pauli) seguido por uma lista de operadores Pauli (X, Y, Z, I) aplicados a qubits específicos. A função `convert_operator`, vista no algoritmo 2, então converte essa lista de operadores para a forma padrão de Jordan-Wigner, criando um produto tensorial onde os operadores Pauli são aplicados aos qubits corretos e o restante é preenchido com a identidade (I). O resultado final é uma lista de termos Hamiltonianos na forma de `SparsePauliOp`, que pode ser usada para simulações quânticas utilizando Qiskit ou rodar em computadores quânticos da IBM.

Algoritmo 2: Função para converter operadores Pauli do formato da PennyLane para o do Qiskit

```

1 def convert_operator(operator, num_qubits):
2     parts = operator.split()
3
4     tensor_product = ['I']*num_qubits
5     for part in parts:
6         tensor_product[int(part[1:])] = part[0]
7
8     return ''.join(tensor_product[::-1])

```

Fonte: Elaborado pelo autor.

5.2 Algoritmo Genético para Estado Referencial

O algoritmo genético desenvolvido tem como objetivo selecionar o estado referencial ótimo para o VQE a partir de um ansatz qualquer. A seguir, detalhamos as principais etapas e componentes do algoritmo.

5.2.1 Função de Aptidão

A função de aptidão, implementada na função `fitness`, é responsável por avaliar a qualidade de cada indivíduo dentro da população do algoritmo genético. Cada indivíduo é representado por uma string binária de comprimento n , onde n é o número de qubits no

circuito. Cada bit da string indica se o respectivo qubit está no estado $|0\rangle$ ou $|1\rangle$ no estado referencial inicial. Por exemplo, um bit igual a 1 significa que uma porta X será aplicada ao qubit correspondente, alterando seu estado de $|0\rangle$ para $|1\rangle$. O código para a função de aptidão é mostrado no algoritmo 3:

Algoritmo 3: Função de aptidão do individuo

```

1 def fitness(individual, args):
2     reference_state = QuantumCircuit(len(individual))
3     for i, bit in enumerate(individual):
4         if bit == 1:
5             reference_state.x(i)
6     reference_state.barrier()
7     circuit = reference_state.compose(args[1])
8
9     energy = cf(args[0], circuit, args[2], args[3])
10
11    print(f"Individual: {individual}, Energy: {energy}")
12    return energy

```

Fonte: Elaborado pelo autor.

A função *cf* é a função de custo que calcula o valor esperado (energia) de um Hamiltoniano para um dado circuito quântico e parâmetros variacionais. O código da função *cf* é apresentado abaixo no algoritmo 4:

Algoritmo 4: Função de custo de um circuito variacional

```

1 def cf(params, ansatz, hamiltonian, estimator):
2     pub = (ansatz, [hamiltonian], [params])
3     energy = estimator.run(pubs=[pub]).result()[0].data.evs[0]
4     return energy

```

Fonte: Elaborado pelo autor.

A função recebe quatro parâmetros principais:

- *params*: Um vetor contendo os parâmetros variacionais que serão aplicados ao *ansatz*.
- *ansatz*: O circuito quântico que prepara o estado quântico variacional com base nos parâmetros fornecidos.
- *hamiltonian*: O Hamiltoniano da molécula ou sistema quântico, que será utilizado para calcular a energia esperada.
- *estimator*: O estimador que calcula a expectativa de valor da energia para o *ansatz* dado.

Dentro da função, um objeto *primitive unified bloc* (PUB) é criado, que contém o *ansatz*, o Hamiltoniano e os parâmetros variacionais. Esse objeto é passado para o *estimator*,

que executa o cálculo da expectativa de valor (energy expectation value) sobre o circuito quântico definido pelo ansatz e parâmetros. O valor esperado da energia é extraído e retornado como resultado da função.

5.2.2 Algoritmo Genético

O algoritmo genético é implementado na função `genetic_algorithm`, que é responsável por gerenciar a evolução da população ao longo de várias gerações. O código completo do algoritmo é mostrado abaixo:

Algoritmo 5: Função do algoritmo genético

```

1 def genetic_algorithm(num_generations = 5, population_size = 10,
2   mutation_rate = 0.1, args = None):
3     n = args[1].num_qubits
4     population = np.random.randint(2, size=(population_size, n))
5     best_individual = None
6     best_fitness = float('inf')
7     for generation in range(num_generations):
8         fitness_scores = [fitness(ind, args) for ind in population]
9         current_min_fitness = min(fitness_scores)
10        if current_min_fitness < best_fitness:
11            best_fitness = current_min_fitness
12            best_individual = population[np.argmin(fitness_scores)]
13        sorted_population = [x for _, x in sorted(zip(
14            fitness_scores, population), key=lambda pair: pair[0])]
15        population = sorted_population[:population_size // 2]
16        next_generation = []
17        while len(next_generation) < population_size:
18            parent1 = random.choice(population)
19            parent2 = random.choice(population)
20            crossover_point = random.randint(1, len(parent1) - 1)
21            child = np.concatenate((parent1[:crossover_point],
22                parent2[crossover_point:]))
23            next_generation.append(child)
24        for individual in next_generation:
25            if random.random() < mutation_rate:
26                mutation_index = random.randint(0, len(individual)
27                    - 1)
28                individual[mutation_index] += random.uniform(-1, 1)
29        population = next_generation
30    return best_individual

```

Fonte: Elaborado pelo autor.

O algoritmo 5 da função `genetic_algorithm`, começa inicializando uma população de indivíduos, onde cada indivíduo é uma *string* binária gerada aleatoriamente, representando

diferentes estados referenciais iniciais. A variável n representa o número de qubits no circuito, e a população inicial tem tamanho definido por `population_size`. Em cada geração, a aptidão de todos os indivíduos é avaliada usando a função `fitness`, e o melhor indivíduo, ou seja, aquele com a menor energia, é armazenado.

Após avaliar a aptidão de todos os indivíduos em uma geração, a população é ordenada com base nos valores de aptidão, e os melhores indivíduos (metade superior da população) são selecionados como "pais" para a próxima geração. O algoritmo então realiza o cruzamento (*crossover*) entre pares de pais, gerando novos indivíduos (filhos) ao combinar partes dos pais. Esse processo continua até que a nova geração tenha o mesmo tamanho da população original. Essa abordagem permite que o VQE inicie em uma configuração que minimiza a energia do sistema, aumentando as chances de convergência para o verdadeiro estado fundamental.

5.3 Ansatz com Portas de Excitações Únicas

A implementação das portas de excitações únicas utilizadas no trabalho é fundamental para a construção do *ansatz* candidato, permitindo a simulação eficiente de estados quânticos moleculares no contexto do VQE. A seguir, é descrito em detalhes o código utilizado para implementar essas portas e o circuito quântico variacional.

5.3.1 Porta $A(\theta, \phi)$

A porta simula as excitações únicas no sistema quântico no espaço de Fock, com isso sendo uma peça fundamental do circuito variacional. A implementação em Python é apresentada pelo algoritmo 6:

Algoritmo 6: Porta de Excitação única

```

1 def A(theta, phi):
2     qc = QuantumCircuit(2)
3     qc.cx(1, 0)
4     qc.rz(2*(-phi+np.pi), 1)
5     qc.ry(2*(-theta+np.pi/2), 1)
6     qc.cx(0, 1)
7     qc.rz(2*(phi-np.pi), 1)
8     qc.ry(2*(theta-np.pi/2), 1)
9     qc.cx(1, 0)
10    return qc.to_gate(label=r"$A(\theta, \phi)$")

```

Nesta implementação, a porta $A(\theta, \phi)$ é construída usando operações de CNOT e rotações nos eixos z (rz) e y (ry) da esfera de Bloch. Os parâmetros θ e ϕ controlam as rotações, ajustando a forma como a excitação é aplicada entre os qubits. O circuito resultante é convertido em uma porta personalizada, rotulada como $A(\theta, \phi)$, permitindo sua aplicação eficiente dentro do *ansatz*.

5.3.2 Função *vf_symmetric*

Para construir o *ansatz* variacional, utilizamos a função `vf_symmetric`, que cria um circuito quântico composto por múltiplas camadas de portas $A(\theta, \phi)$. Este circuito é projetado para ser simétrico, o que ajuda a capturar as correlações eletrônicas de maneira eficiente. A implementação é apresentada pelo algoritmo 7.

Algoritmo 7: Função `vf_symmetric` para construção do *ansatz* variacional

```

1 def vf_symmetric(num_qubits = None, num_layers = 1):
2     vf = QuantumCircuit(num_qubits)
3     theta = ParameterVector(r"$\theta$", num_qubits * num_layers)
4     phi = ParameterVector(r"$\phi$", 2 * num_layers)
5     idx_theta = 0
6     idx_phi = 0
7     for i in range(num_layers):
8         for j in range(num_qubits - 1):
9             if j % 2 == 0:
10                vf.append(A(theta[idx_theta], phi[idx_phi]), [j, j
11                           +1])
12                idx_theta += 1
13     return vf

```

Fonte: Elaborado pelo autor.

A função `vf_symmetric` constrói o circuito variacional utilizando várias instâncias das portas $A(\theta, \phi)$, organizando os parâmetros θ e ϕ em vetores de parâmetros (`ParameterVector`). Em cada camada do circuito, as portas são aplicadas alternadamente em pares de qubits adjacentes, garantindo que a simetria desejada no *ansatz* seja preservada. Essa abordagem modular não só simplifica a construção dos circuitos, como também mantém a fidelidade aos aspectos simétricos da molécula, permitindo uma exploração mais precisa do espaço de Fock.

6 RESULTADOS

Este capítulo apresenta os resultados obtidos neste trabalho. Serão abordados, em sequência, o Hamiltoniano do MoO₃, o estado referencial utilizado, o circuito variacional desenvolvido, os resultados das simulações realizadas tanto em ambientes ideais (sem ruído) quanto em cenários com ruído, e, por fim, os resultados obtidos a partir da execução em hardware quântico real.

6.1 Hamiltoniano do Trióxido de Molibdênio

O Hamiltoniano da molécula MoO₃ obtido é demonstrado na tabela, resultando em um total de 2239 termos. A Tabela 2 apresenta alguns dos termos mais representativos do Hamiltoniano. Observa-se que o termo de maior magnitude corresponde à identidade completa (IIIIIIIIIIIIIIIIIIII), com um coeficiente de -68.026203 , indicando uma contribuição significativa para o valor energético da molécula. Embora termos com coeficientes menores possam parecer menos influentes, cada termo do Hamiltoniano desempenha um papel crucial na busca pela acurácia química na determinação da energia fundamental da molécula.

6.2 Circuito Variacional Simétrico

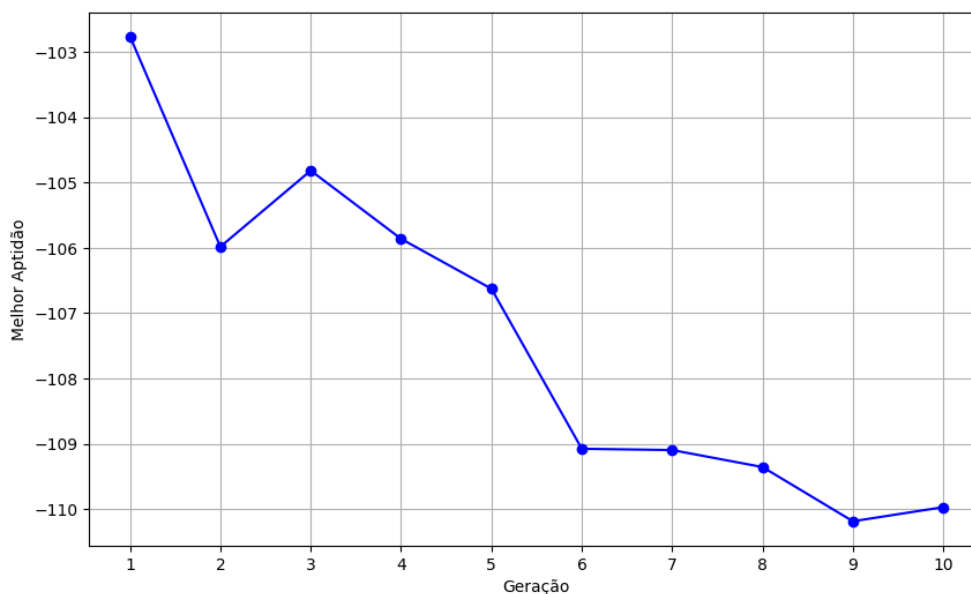
O estado referencial inicial para o circuito foi obtido por meio de um algoritmo genético, executado ao longo de 10 gerações com uma população de 50 indivíduos. O melhor resultado foi identificado no indivíduo [11111111111111110000], o qual apresentou uma energia de -109.91889968256868 . A evolução dos valores de energia ao longo das gerações está ilustrada na Figura 13.

O resultado dessa otimização proporcionou um estado inicial favorável, que foi combinado com o *ansatz* para a obtenção do circuito variacional simétrico final mostrado na figura 14.

6.3 Simulações

As simulações foram organizadas em três categorias: circuitos variacionais, otimizadores e a simulação final com ruído. Essa estrutura permite a busca pelo melhor *workflow* do VQE para o MoO₃, otimizando cada etapa de forma a garantir tanto a eficiência computacional

Figura 13: Otimização do algoritmo genético



Fonte: Elaborado pelo próprio autor

quanto a precisão dos resultados.

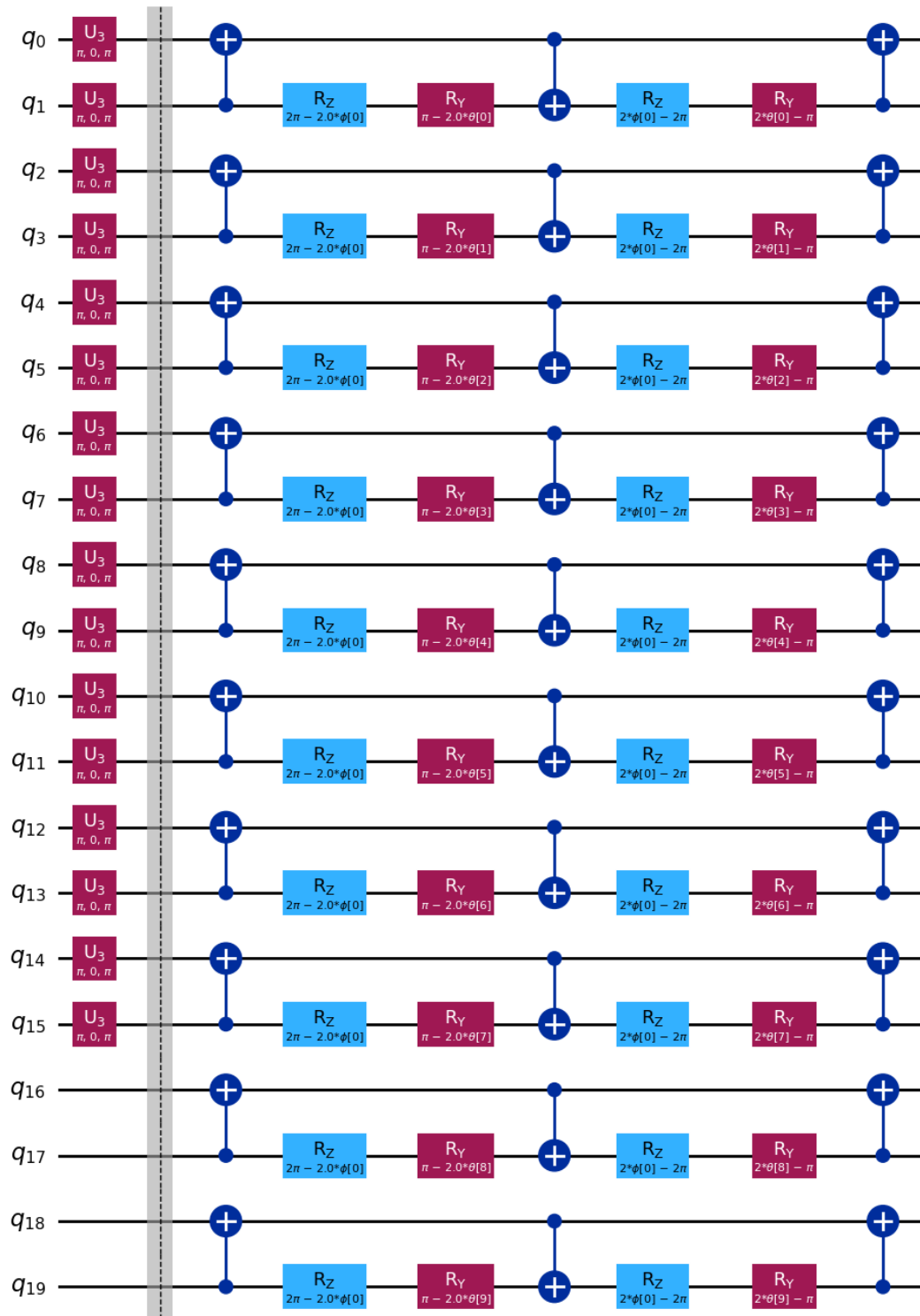
6.3.1 Benchmark de Circuitos Variacionais

A Figura 15 mostra a convergência do algoritmo VQE para a molécula de MoO_3 em uma simulação sem ruído, utilizando diferentes circuitos variacionais: *full*, *linear*, *reverse linear*, *pairwise*, *circular*, *shifted-circular-alternating* (*sca*) e *symmetric*, o proposto por esse trabalho. O otimizador COBYLA foi empregado para conduzir as otimizações, com um limite de 50 iterações. A escala logarítmica foi utilizada para facilitar a visualização das diferenças de desempenho entre os circuitos.

Dentre todos os circuitos testados, apenas o circuito *symmetric* conseguiu alcançar a zona de acurácia química, demonstrando uma clara superioridade em termos de precisão e estabilidade ao longo das iterações. Os demais circuitos, embora estáveis, falharam em atingir a acurácia necessária dentro do limite de 50 iterações, com alguns, como o circuito *sca*, apresentando maior variabilidade e uma convergência significativamente mais lenta. Esses resultados indicam que o circuito *symmetric* é particularmente eficaz para simulações VQE em moléculas complexas, especialmente quando utilizando otimizadores que não são os mais precisos, como o COBYLA.

Os gráficos de características dos circuitos, apresentados nas Figuras 16a, 16b e 16c, revelam que o circuito *full* possui a maior profundidade (Figura 16a) e o maior número de

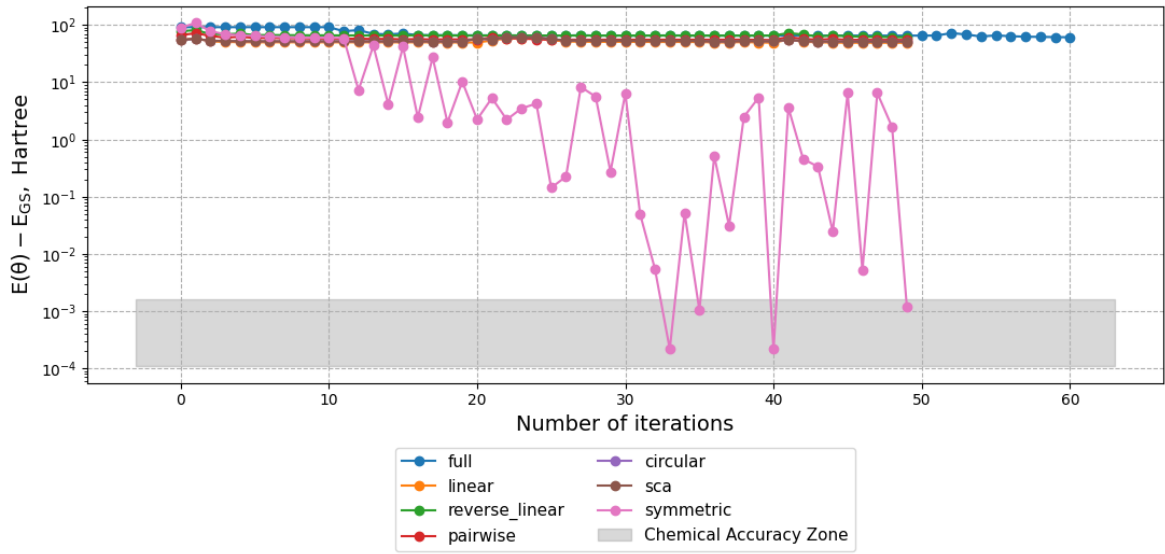
Figura 14: Circuito Variacional



Fonte: Elaborado pelo próprio autor

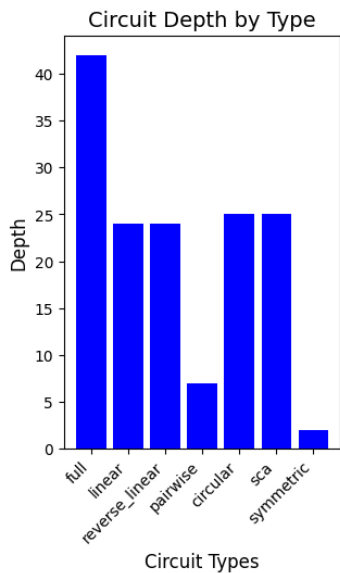
operações multi-qubit (Figura 16b), o que pode explicar sua falta de eficiência em convergir para a zona de acurácia química, apesar de seu potencial teórico. A Figura 16c mostra a distribuição total de operações por tipo de circuito, reforçando o impacto das características mencionadas

Figura 15: Convergência para diferentes circuitos variacionais
MoO₃ Convergence - Noiseless Simulation

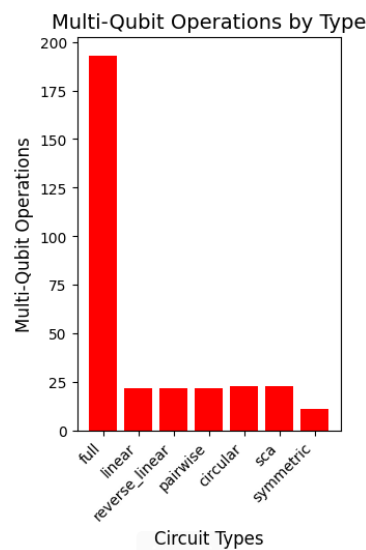


Fonte: Elaborado pelo próprio autor

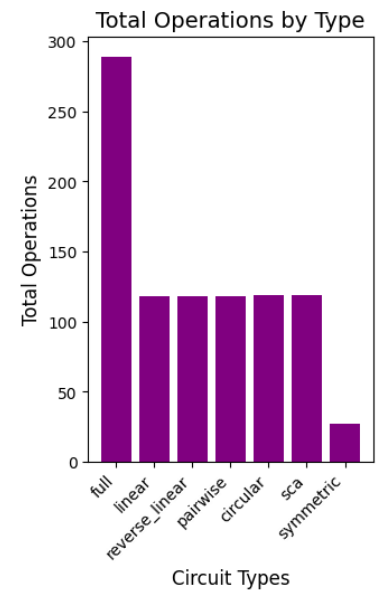
anteriormente. Por outro lado, o circuito proposto *symmetric* destacou-se não apenas por atingir a acurácia necessária, mas também por apresentar uma complexidade significativamente menor em termos de profundidade e operações totais. Este equilíbrio entre simplicidade estrutural e eficiência no processo de otimização torna o circuito *symmetric* uma escolha promissora para simulações VQE, particularmente quando a estabilidade e precisão são essenciais.



(a) Distribuição da profundidade dos circuitos por tipo



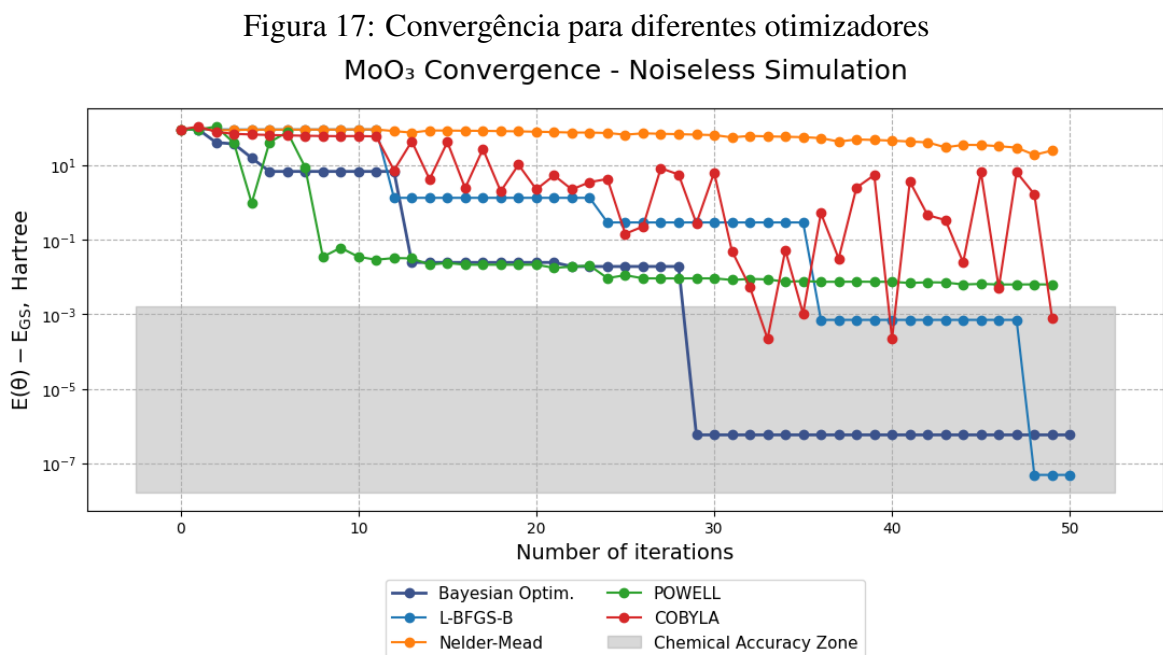
(b) Distribuição das operações multi-qubit por tipo de circuito
Fonte: Elaborado pelo próprio autor



(c) Distribuição total de operações por tipo de circuito

6.3.2 Benchmark de Otimizadores

A Figura 17 apresenta a convergência do algoritmo VQE para a molécula de MoO_3 em uma simulação sem ruído, utilizando diferentes otimizadores: L-BFGS-B, Nelder-Mead, POWELL, COBYLA e Bayesian Optimization. A análise da convergência foi realizada em 50 iterações, com o objetivo de atingir uma acurácia química de 0.0016 Hartrees, a escala utilizada é logarítmica. Os otimizadores L-BFGS-B e Bayesian Optimization demonstraram o melhor desempenho, atingindo rapidamente a zona de acurácia química e mantendo estabilidade ao longo das iterações.



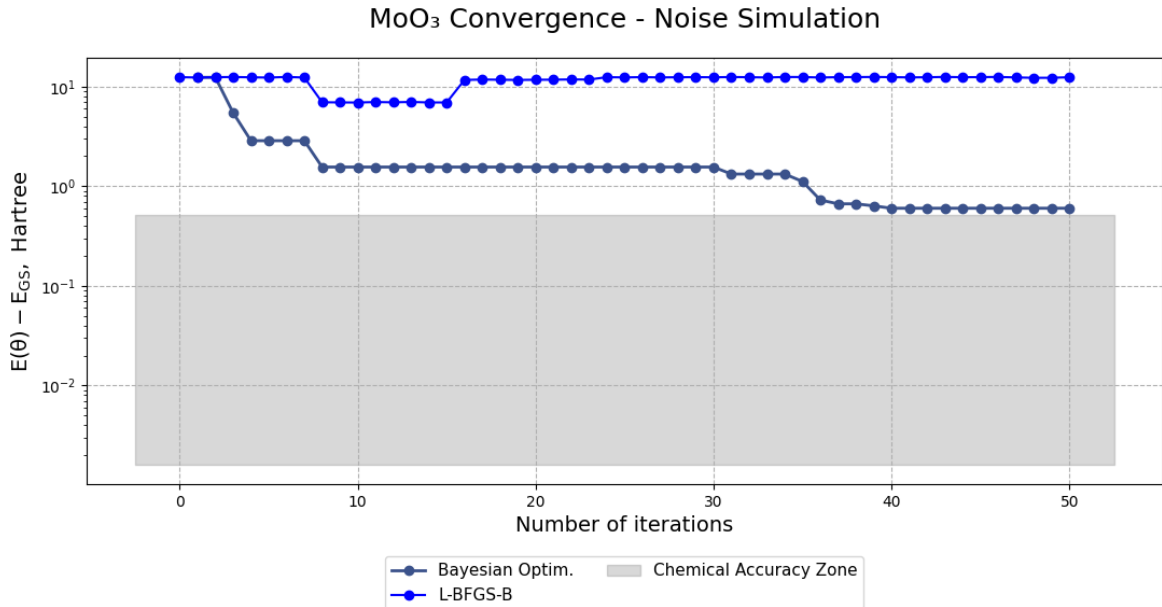
Os otimizadores Nelder-Mead e COBYLA apresentaram maior variabilidade e menor precisão, falhando em alcançar a acurácia desejada dentro das iterações permitidas. O otimizador POWELL, embora estável, também não atingiu a precisão necessária. Esses resultados destacam a eficácia dos otimizadores L-BFGS-B e Bayesian Optimization na simulação de algoritmos quânticos para moléculas complexas, como o MoO_3 .

6.3.3 Simulação com Ruído

A Figura 18 apresenta os resultados da simulação com ruído, utilizando os otimizadores L-BFGS-B e Bayesian Optimization, previamente identificados como os mais eficazes na simulação sem ruído. A simulação foi realizada no processador *Eagle r3* do *backend* *ibm_kyiv*,

incorporando os efeitos do ruído característicos do *hardware* quântico, como erros nas portas ECR, SX e X.

Figura 18: Convergência dos otimizadores utilizando simulação com ruído



Fonte: Elaborado pelo próprio autor

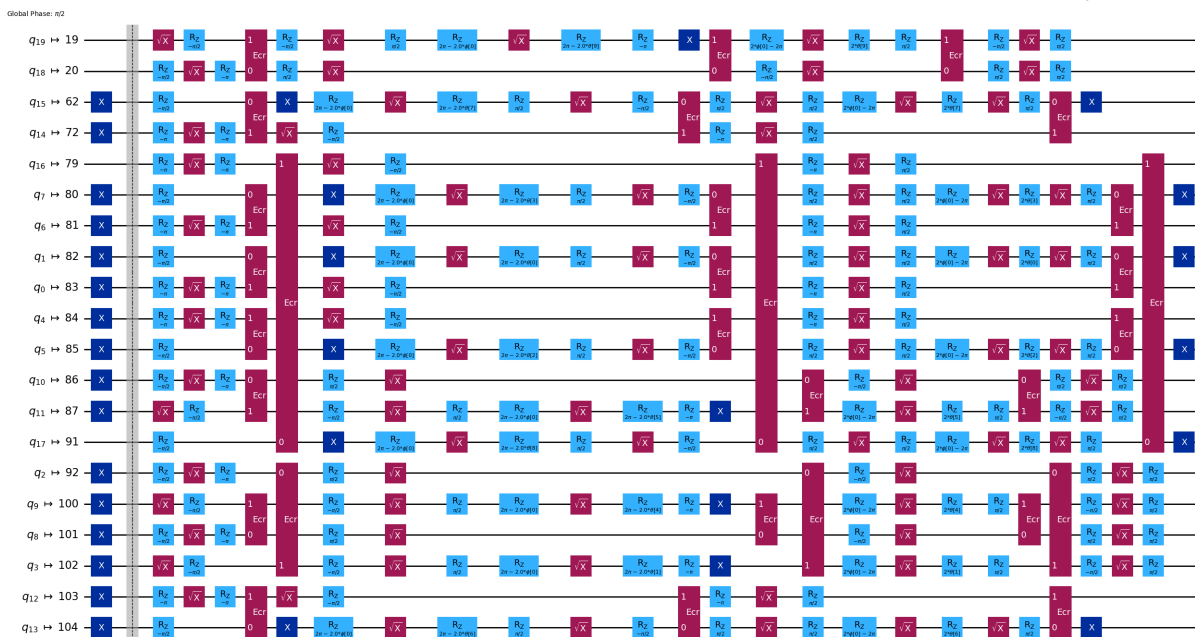
Os resultados indicam que o otimizador Bayesian Optimization foi o mais robusto, demonstrando uma convergência mais estável porém não tão precisa para entrar dentro da condição de acurácia química, alcançando valores próximos. O L-BFGS-B, por outro lado, apresentou maior variabilidade nos resultados, sugerindo que o impacto do ruído é mais significativo para esse método, o que pode estar relacionado à sua natureza exploratória e dependente de modelos probabilísticos. Embora ambos os otimizadores tenham sido afetados pelo ruído, o Bayesian Optimization se destacou como a escolha mais confiável para execuções em hardware real, onde a mitigação de ruído é limitada.

6.4 Execução em Hardware Quântico

Após a realização das simulações com e sem ruído, onde os circuitos variacionais e otimizadores foram testados, a última etapa consistiu na execução em hardware quântico real. Utilizou-se o processador *Eagle r3* do *ibm_kyiv*, equipado com 127 qubits, para implementar o circuito variacional selecionado em conjunto com o otimizador Bayesian Optimization. A transpilação do circuito foi feita através da biblioteca `qiskit.transpiler.preset_passmanagers`, assegurando a adaptação precisa das portas quânticas ao hardware como pode ser visto na figura

19, bem como a formatação correta do Hamiltoniano.

Figura 19: Circuito variacional transpilado para rodar no *hardware* alvo *ibm_kyiv*

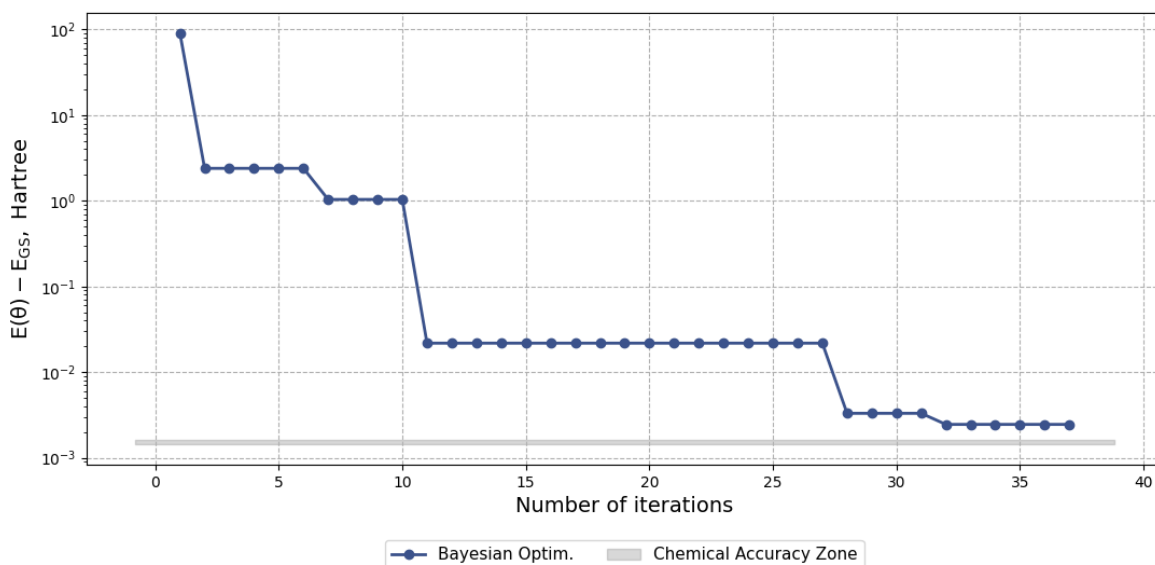


Fonte: Elaborado pelo próprio autor

Para mitigar os efeitos do ruído inerente ao hardware, foram aplicadas técnicas de desacoplamento dinâmico (`DynamicalDecouplingOptions`) e extrapolação de ruído zero (ZNE), fundamentais para a melhoria da precisão durante a execução.

Figura 20: Convergência do Bayesian no *hardware* *ibm_kyiv*

MoO₃ Convergence - Real Hardware



Fonte: Elaborado pelo próprio autor

O resultado na Figura 20 mostra que foi possível executar apenas 37 iterações no hardware, em vez das 50 inicialmente planejadas. Embora o gráfico demonstre uma convergência eficiente, o algoritmo não conseguiu atingir a zona de acurácia química dentro das iterações realizadas. No entanto, observa-se uma melhoria significativa em comparação com a simulação com ruído, onde a convergência ocorreu de forma menos precisa. É possível que, com um tempo de execução superior a 10 minutos, o resultado pudesse ter sido ainda mais aprimorado.

Tabela 2: Termos amostrados do Hamiltoniano da molécula MoO₃.

Index	Pauli String	Coefficient
0	IIIIIIIIIIIIIIIIIIII	-68.026203
1	IZIIIIIIIIIIIIIIIIII	0.027739
2	ZIIIIIIIIIIIIIIIIII	0.027739
3	IIIIZIIIIIIIIIIIIIIII	2.102205
4	IIIZIIIIIIIIIIIIIIII	2.102205
...		
2234	XZZZZZZZYZZZZZZZZZZX	0.003502
2235	YZZZZZZZZZZZZZZZZZZXXZZY	0.006820
2236	XZZZZZZZZZZZZZZZZZZYYZZX	0.006820
2237	XZZZZZZZZZZZZZZZZZZXYY	0.037064
2238	YZZZZZZZZZZZZZZZZZZYXX	0.037064

Fonte: Elaborado pelo próprio autor

7 CONCLUSÃO E TRABALHOS FUTUROS

Com base nos resultados apresentados, conclui-se que o algoritmo VQE, quando aplicado à molécula de MoO_3 , demonstrou uma eficácia considerável, especialmente quando combinado com as estratégias criadas neste trabalho como: utilizar de um circuito inicial definido por algoritmo genético, circuito variacional simétrico simplificado e a otimização bayesiana. Dentre os diversos circuitos testados, o circuito *symmetric* destacou-se por alcançar a zona de acurácia química durante as simulações em ambiente sem ruído, demonstrando uma excelente relação entre simplicidade estrutural e eficiência de convergência. Esse resultado sugere que o circuito *symmetric* é uma escolha promissora para simulações envolvendo moléculas complexas, onde tanto a estabilidade quanto a precisão são cruciais.

Na avaliação dos otimizadores, o Bayesian Optimization provou ser o mais robusto, tanto em simulações com ruído quanto na execução em hardware real. Embora o ruído tenha impactado os resultados, o otimizador manteve sua eficácia, mostrando uma convergência estável e alcançando valores próximos à acurácia química, mesmo em condições adversas. Em contraste, o otimizador L-BFGS-B, embora eficiente em ambientes sem ruído, demonstrou maior sensibilidade ao ruído, o que pode limitar sua aplicabilidade em execuções em hardware real.

Por fim, a execução no hardware quântico real, embora limitada a 37 iterações devido ao tempo de execução restrito, confirmou a capacidade de ambos o circuito e o otimizador Bayesian Optimization de operar de maneira eficiente em um ambiente quântico prático. Apesar de não atingir a zona de acurácia química, a melhoria observada em relação à simulação com ruído sugere que, com mais tempo de execução, resultados ainda mais precisos poderiam ser obtidos.

7.1 Trabalhos Futuros

Os trabalhos futuros que podem ser realizados incluem a exploração de diferentes bases além da STO- n G, como as bases de maior precisão, por exemplo, cc-pVDZ ou cc-pVTZ. A utilização dessas bases pode proporcionar resultados mais exatos, permitindo uma descrição mais fiel da molécula de MoO_3 e, conseqüentemente, melhorando a acurácia das simulações de energia. Além disso, a metodologia desenvolvida neste trabalho pode ser aplicada para a obtenção de estados de energia excitados, expandindo o escopo das simulações quânticas para além do estado fundamental.

REFERÊNCIAS

- AMINIAN, M.; SAEEDI, M.; ZAMANI, M. S.; SEDIGHI, M. Fpga-based circuit model emulation of quantum algorithms. In: IEEE COMPUTER SOCIETY ANNUAL SYMPOSIUM ON VLSI, 2008, Montpellier, France. **IEEE Computer Society Annual Symposium on VLSI**. Montpellier: IEEE, 2008. p. 399–404.
- BLUNT, N. S.; CAMPS, J.; CRAWFORD, O.; IZAK, R.; LEONTICA, S.; MIRANI, A.; MOYLETT, A. E.; SCIVIER, S. A.; SUNDERHAUF, C.; SCHOPF, P.; TAYLOR, J. M.; HOLZMANN, N. Perspective on the current state-of-the-art of quantum computing for drug discovery applications. **Journal of Chemical Theory and Computation**, American Chemical Society (ACS), v. 18, n. 12, p. 7001–7023, nov 2022. Disponível em: <https://doi.org/10.1021/acs.jctc.2c00574>. Acesso em: 07 nov. 2023.
- BORN, M.; OPPENHEIMER, R. Zur quantentheorie der molekeln. **Annalen der Physik**, Wiley Online Library, v. 389, n. 20, p. 457–484, 1927.
- BOYD, S.; VANDENBERGHE, L. **Convex Optimization**. Cambridge: Cambridge University Press, 2004.
- BYRD, R. H.; LU, P.; NOCEDAL, J.; ZHU, C. A limited memory algorithm for bound constrained optimization. **SIAM Journal on Scientific Computing**, v. 16, n. 5, p. 1190–1208, 1995. Disponível em: <https://doi.org/10.1137/0916069>. Acesso em: 20 jul. 2024.
- CAO, Y.; ROMERO, J.; OLSON, J. P.; DEGROOTE, M.; JOHNSON, P. D.; KIEFEROVÁ, M.; KIVLICHAN, I. D.; MENKE, T.; PEROPADRE, B.; SAWAYA, N. P. D.; SIM, S.; VEIS, L.; ASPURU-GUZIĆ, A. Quantum chemistry in the age of quantum computing. **Chemical Reviews**, v. 119, n. 19, p. 10856–10915, 2019. PMID: 31469277. Disponível em: <https://doi.org/10.1021/acs.chemrev.8b00803>. Acesso em: 03 out. 2023.
- CHUANG, I. L.; YAMAMOTO, Y. Simple quantum computer. **Physical Review A**, American Physical Society (APS), v. 52, n. 5, p. 3489–3496, nov 1995. Disponível em: <https://doi.org/10.1103/physreva.52.3489>. Acesso em: 28 ago. 2023.
- COHEN-TANNOUJDI, C.; LALOË, F.; DIU, B. **Quantum mechanics Vol 3: Fermions, bosons, photons, correlations, and entanglement**. Germany: Wiley VCH, 2019.
- DEUTSCH, D.; PENROSE, R. Quantum theory, the church–turing principle and the universal quantum computer. **Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences**, v. 400, n. 1818, p. 97–117, 1985. Acesso em: 4 set. 2023. Disponível em: <https://royalsocietypublishing.org/doi/abs/10.1098/rspa.1985.0070>. Acesso em: 20 set. 2023.
- FEDOROV, D. A.; PENG, B.; GOVIND, N.; ALEXEEV, Y. **VQE Method: A Short Survey and Recent Developments**. 2021.
- GARD, B. T.; ZHU, L.; BARRON, G. S.; MAYHALL, N. J.; ECONOMOU, S. E.; BARNES, E. Efficient symmetry-preserving state preparation circuits for the variational quantum eigensolver algorithm. **npj Quantum Information**, Nature Publishing Group, v. 6, p. 10, 2020. Disponível em: <https://doi.org/10.1038/s41534-019-0240-1>. Acesso em: 12 jul. 2023.
- GRIFFITHS, D. J.; SCHROETER, D. F. **Introduction to Quantum Mechanics**. 3rd. ed. Cambridge: Cambridge University Press, 2018.

GROVER, L. K. **A fast quantum mechanical algorithm for database search**. 1996. Disponível em: <https://arxiv.org/abs/quant-ph/9605043>. Acesso em: 10 out. 2023.

HANER, T.; STEIGER, D. S.; SMELYANSKIY, M.; TROYER, M. High performance emulation of quantum circuits. In: SC16 INTERNATIONAL CONFERENCE FOR HIGH PERFORMANCE COMPUTING, NETWORKING, STORAGE AND ANALYSIS, 2016, Salt Lake City, USA. **Proceedings of the SC16: International Conference for High Performance Computing, Networking, Storage and Analysis**. Salt Lake City: IEEE, 2016. p. 866–874. Disponível em: <https://doi.org/10.1109/sc.2016.73>. Acesso em: 23 ago. 2023.

JAVADI-ABHARI, A.; TREINISH, M.; KRSULICH, K.; WOOD, C. J.; LISHMAN, J.; GACON, J.; MARTIEL, S.; AL. et. Quantum computing with qiskit. **arXiv**, May 2024. Disponível em: <https://arxiv.org/abs/2405.08810>. Acesso em: 23 ago. 2024.

JORDAN, P.; WIGNER, E. Über das paulische Äquivalenzverbot. **Zeitschrift für Physik**, v. 47, p. 631–651, 1928. Disponível em: <https://doi.org/10.1007/BF01331938>. Acesso em: 16 jun. 2024.

KHALID, A. U.; ZILIC, Z.; RADECKA, K. Fpga emulation of quantum circuits. In: IEEE, ICCD 2004., 2004, San Jose, United States. **IEEE International Conference on Computer Design: VLSI in Computers and Processors**. San Jose: IEEE, 2004. p. 310–315.

KUREK, M. **Quantum Technologies: Patents, Publications Investissements Landscape**. [S. l.], 2020.

LAU, J.; LIM, K.; SHROTRIYA, H. *et al.* Nisq computing: where are we and where do we go? **AAPPS Bull.**, v. 32, p. 27, 2022. Disponível em: <https://doi.org/10.1007/s43673-022-00058-z>. Acesso em: 10 set. 2023.

LIU, P. **Bayesian Optimization: Theory and Practice Using Python**. 1. ed. [S. l.]: Apress Berkeley, CA, 2023. XV, 234 p. Published: 24 March 2023. ISBN 978-1-4842-9062-0.

MANENTI, R.; MOTTA, M. **Quantum Information Science**. Oxford University Press, 2023. ISBN 9780198787488. Disponível em: <https://books.google.com.br/books?id=UkbMEAAAQBAJ>. Acesso em: 20 jul. 2024.

NELDER, J. A.; MEAD, R. A Simplex Method for Function Minimization. **The Computer Journal**, v. 7, n. 4, p. 308–313, 01 1965. ISSN 0010-4620. Disponível em: <https://doi.org/10.1093/comjnl/7.4.308>.

NIELSEN, M.; CHUANG, I. **Quantum Computation and Quantum Information: 10th Anniversary Edition**. [S. l.]: Cambridge University Press, 2010.

PATEL, A. The quantum density matrix and its many uses. **Journal of the Indian Institute of Science**, v. 103, p. 401–417, 2023.

PERUZZO, A.; MCCLEAN, J.; SHADBOLT, P.; YUNG, M.-H.; ZHOU, X.-Q.; LOVE, P. J.; ASPURU-GUZZIK, A.; O'BRIEN, J. L. A variational eigenvalue solver on a photonic quantum processor. **Nature Communications**, Springer Science and Business Media LLC, v. 5, n. 1, jul 2014. Disponível em: <https://doi.org/10.1038/ncomms5213>.

POWELL, M. J. D. An efficient method for finding the minimum of a function of several variables without calculating derivatives. **The Computer Journal**, v. 7, n. 2, p. 155–162, 01 1964. ISSN 0010-4620. Disponível em: <https://doi.org/10.1093/comjnl/7.2.155>. Acesso em: 12 jul. 2024.

POWELL, M. J. D. A direct search optimization method that models the objective and constraint functions by linear interpolation. **Advances in Optimization and Numerical Analysis**, v. 275, p. 51–67, 1994. Disponível em: https://doi.org/10.1007/978-94-015-8330-5_4.

RAMEZANI, S. B.; SOMMERS, A.; MANCHUKONDA, H. K.; RAHIMI, S.; AMIRLATIFI, A. Machine learning algorithms in quantum computing: A survey. In: INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS, 2020, Glasgow, UK. **International Joint Conference on Neural Networks (IJCNN)**. Glasgow: IEEE, 2020. p. 1–8.

ROCHA, L. Viana Maurat da; SILVA, P. S. Rangel Cruz da; SOUZA, D. de H. S.; TAVARES, M. I. B. Molybdenum trioxide (moo3): a scoping review of its properties, synthesis and applications. **Concilium**, v. 24, n. 6, p. 443–462, 2024. Disponível em: <https://doi.org/10.53660/CLM-3190-24F41>. Acesso em: 5 ago. 2024.

SHANKLAND, S. **Intel enters the quantum computing horse race with 12-qubit chip**. 2023. Disponível em: <https://www.cnet.com/tech/computing/intel-enters-the-quantum-computing-horse-race-with-12-qubit-chip/>. Acesso em: 10 set. 2023.

SHOR, P. W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. **SIAM Journal on Computing**, Society for Industrial & Applied Mathematics (SIAM), v. 26, n. 5, p. 1484–1509, oct 1997. Disponível em: <https://doi.org/10.1137/S0097539795293172>. Acesso em: 12 set. 2023.

TAZI, C. L.; THOM, A. J. W. Folded spectrum vqe: A Quantum Computing Method for the Calculation of Molecular Excited States. **Journal of Chemical Theory and Computation**, v. 20, n. 6, p. 2491–2504, 2024. PMID: 38492238. Disponível em: <https://doi.org/10.1021/acs.jctc.3c01378>. Acesso em: 30 jun. 2024.

THAKKAR, A. J. **Quantum Chemistry (Third Edition)**. IOP Publishing, 2021. (2053-2563). ISBN 978-0-7503-3827-1. Disponível em: <https://dx.doi.org/10.1088/978-0-7503-3827-1>. Acesso em: 5 jul. 2024.

THOMPSON, N.; SPANUTH, S. **The Decline of Computers As a General Purpose Technology: Why Deep Learning and the End of Moore’s Law are Fragmenting Computing**. 2018. <http://dx.doi.org/10.2139/ssrn.3287769>. Disponível em SSRN: <https://ssrn.com/abstract=3287769>.

WAIDYASOORIYA, H. M.; OSHIYAMA, H.; KUREBAYASHI, Y.; HARIYAMA, M.; OHZEKI, M. A scalable emulator for quantum fourier transform using multiple-fpgas with high-bandwidth-memory. **IEEE Access**, v. 10, p. 65103–65117, 2022.

WIERICHS, D.; GOGOLIN, C.; KASTORYANO, M. Avoiding local minima in variational quantum eigensolvers with the natural gradient optimizer. **Physical Review Research**, v. 2, n. 4, p. 043246, Nov 2020. Disponível em: <https://link.aps.org/doi/10.1103/PhysRevResearch.2.043246>. Acesso em: 9 ago. 2024. Disponível em: <https://link.aps.org/doi/10.1103/PhysRevResearch.2.043246>.

WITT, S. **The world-changing race to develop the quantum computer.** 2022. Disponível em: <https://www.newyorker.com/magazine/2022/12/19/the-world-changing-race-to-develop-the-quantum-computer>. Acesso em: 12 set. 2023.

ZAGOSKIN, A. **Quantum Engineering:** Theory and Design of Quantum Coherent Structures. [S. l.]: Cambridge University Press, 2011.