Development of Symmetry-Preserving Variational Quantum Algorithms for Solving Many-body Problems

Hamzat Adebayo Akande

December, 2024

Supervised By: Professor Matthieu Saubanère

This master's thesis is submitted to the ICTP-East African Institute for Fundamental Research, University of Rwanda, for partial fulfillment of the requirement of a Master's degree in condensed matter physics

Preface

This dissertation, *Development of Symmetry-Preserving Variational Quantum Algorithms for Solving Many-body Problems*, takes up those findings and methodologies that proved to be essential for building the Variational Hyperspace Projection (VHyP) algorithm.

Part of the present work has been submitted to the archive under the title *"Symmetry-preserved cost functions for variational quantum eigensolver"* by Hamzat Akande, Bruno Senjean, and Matthieu Saubanere. This can be found on arXiv: arXiv:2411.16915 [quant-ph].

The manuscript details the theoretical framework and results, focusing on the symmetry-preserving cost functions that generalize the standard Variational Quantum Eigensolver (VQE).

Abstract

This work focuses on developing efficient quantum algorithms for solving many-body quantum problems by implementing variational quantum eigensolver (VQE) associated with shallow quantum circuits without optimization issues such as Barren plateaus. This research aims to overcome the disadvantages of hardware-efficient ansatz (HEA) through a novel approach that would incorporate circuit optimization methods together with symmetry-preserving techniques.

We propose to modify the standard VQE algorithm to compute materials, molecules, and electronic structures on a quantum computing simulator. The decomposition of the diagonalization of the Hamiltonian into block-diagonalized form with Householder transformation allows the define symmetry-preserving cost functions and to setup iterative processes. The resulting circuits become shallower, and optimization is simplified. Numerical results show in detail the effectiveness of this solver by showing shallower circuits and achieving higher performances compared to the standard VQE approaches.

We implement and test the algorithm for the ground-state properties of H_2, LiH, HeH^+, H_3^- and H_4. We compared various variational approaches to the fidelity of the results with full configuration interaction (FCI) and showed that the method introduced here can surpass the standard VQE in both accuracy and efficiency. These advancements offer a promising path for achieving practical quantum simulations on near-term quantum devices.

Keywords: Quantum algorithms, Variational Quantum Eigensolver (VQE), Householder transformation, Circuit optimization, Full Configuration Interaction (FCI).

Contents

List of Figures

List of Tables

1 Introduction

1.1 Background and Motivation

Understanding the essential nature of matter at the atomic and molecular levels is a fundamental aspect of modern science and technology. Such understanding makes it possible to create and improve materials with properties designed to fit the specific needs of applications in areas such as energy, medicine, and engineering. Scientists can predict material properties, chemical reactions, and structural dynamics using theoretical frameworks and computational modeling, eliminating the need for expensive and time-consuming trial-and-error experiments. These predictive capabilities have revolutionized materials science, allowing for enormous improvements in a variety of fields, including renewable energy innovations, medicinal developments, and improved industrial processes.

Classical computing systems have been used extensively in a wide range of theoretical calculations to predict the properties of materials for the last few decades. Nevertheless, the shortcomings emerge when using these computers to imitate large quantum systems, which are relevant to, for instance, material science, energy storage, or pharmaceutical development. The classical computing's main problem is associated with the exponential increase of the Hilbert space as the quantum particles, e.g., electrons, increase in number. To be specific, for an electron system with N spin orbitals, the total number of quantum states increases by a factor of 2^N, and it is simply impractical for classical computers to keep and manipulate all the different states store and manipulate all possible states as N goes up. Such a scaling problem will then present a big obstacle in the accurate replication of systems having a large number of electrons.

While these classical methods yield useful approximations, they often cannot capture the full complexity of large quantum systems. The new paradigm provided by quantum computing is a dramatically new alternative that exploits the principles of quantum mechanics to represent and manipulate quantum states with far greater ease [2, 3, 4]. Theoretical frameworks do propose that quantum computers have the potential to solve many-body problems by manipulating qubits, which naturally operate in a superposition of states. This property helps to avoid the exponential growth of Hilbert space.

This section explores motivations for quantum computing, a description of how it works, and the problems it faces in becoming a useful computational tool.

1.2 The Principles of Quantum Computing

Quantum computers use quantum bits, or qubits. Qubits can exist in a superposition of states, entanglement, and interference, enabling them to process and store exponentially more information than classical bits. A digital quantum computer uses quantum gates, which are the quantum analogs of classical logic gates, to make operations. Various gates, when applied together, form a quantum circuit that performs calculations.

1.2.1 Challenges in Quantum Computing

The quantum computer has encountered many obstacles although it has a lot of potential:

- **Decoherence and Noise:** The current Noisy Intermediate-scale quantum (NISQ) device is very sensitive to the external environment which leads to decoherence and causes qubits to lose their quantum state.[5].
- **Error Correction:** To overcome the unreliability of quantum computations, error correction is essential, however, it requires plenty of additional physical qubits [6].
- **Scalability:** The construction of large-scale fully operational quantum computers with millions of qubits is a very difficult process that would need a high degree of control and detuning [7, 8].
- **Algorithm Development:** The design of quantum algorithms that are energy-efficient and surpass classical ones is critical for many aspects like computation time, problem sizes, etc. Some notable breakthroughs in quantum computing such as Shor's algorithm for factoring large numbers and Grover's algorithm for searching have been made, yet no unique quantum algorithm potentially solving many NP-complete problems exists till now. So, it is still a challenge to develop quantum algorithms that can speed up many algorithms. [9].

1.3 Quantum Computing for Many-body Problems

The quantum many-body problem, especially in quantum chemistry, is widely regarded as one of the most complex computational challenges in condensed matter physics. It involves solving the Schrödinger equation for interacting electrons to determine the ground state properties of molecular systems. Classical computational techniques, such as diagonalization of the Hamiltonian, are feasible only for small systems. Methods like Full Configuration Interaction (FCI) and Coupled-Cluster Theory (CC) are widely used in quantum chemistry; however, FCI scales exponentially, and CC, though scaling polynomially, remains computationally intensive, making both approaches impractical for large molecules or materials. [10, 11, 12].

Quantum computing, with its ability to naturally simulate quantum systems, offers a promising approach to addressing the quantum many-body problem. Fully quantum algorithms, such as Quantum Phase Estimation (QPE), are powerful tools for determining the eigenvalues (or phases) of a unitary operator [13, 14, 15]. However, QPE faces challenges, including efficient state preparation, circuit depth, and sensitivity to noise. To leverage Noisy Intermediate-Scale Quantum (NISQ) devices, the variational principle was adapted into quantum algorithms, leading to the development of the Variational Quantum Eigensolver (VQE) [1]. VQE is designed to solve optimization problems by preparing and measuring quantum states to determine the ground-state energy of molecules. It stands out as the most thoroughly studied algorithm for near-term quantum devices.

1.3.1 The Evolution of Variational Quantum Eigensolver (VQE)

Variational Quantum Eigensolver (VQE) is a hybrid algorithm. It is composed of a parameterized quantum circuit (also known as an *ansatz*) that prepares trial wavefunctions, and a classical optimizer that iteratively minimizes the expectation value of the Hamiltonian [1]. The hybrid nature of the VQE process comes from the fact that energy measurements are performed on quantum hardware, while the optimization is done on the classical computers [16, 17]. VQE has successfully demonstrated ground-state energy calculations for small molecules like H_2, LiH, and BeH_2 [16, 17, 18]. However, VQE has several limitations that have led to the development of numerous variants over the years.

Physically motivated ansatzes have been proposed to contain symmetries of the Hamiltonian. One well-known example is the Unitary Coupled-Cluster (UCC) ansatz, commonly used in quantum chemistry to preserve symmetries such as particle number and spin [19, 20]. Although advantageous, the UCC ansatz also poses challenges of scalability, since the number of cluster operator terms increases with system size [19]. To mitigate this problem, techniques that iteratively construct the ansatz by choosing vital operators have been introduced such as ADAPT-VQE which decreases circuit depth and CNOT gate count while retaining accuracy in ground state energy calculations[21].

The Hardware-Efficient Ansatz (HEA) used in VQE is designed to be flexible and suitable for near-term quantum devices, making it a valuable tool for practical quantum computing applications [22]. Its simplicity and adaptability allow for efficient implementation on noisy intermediate-scale quantum (NISQ) devices. However, HEA also presents challenges. It often breaks the symmetry of the Hamiltonian, limiting its accuracy in representing molecular systems. HEA circuits are prone to *barren plateau phenomenon*, where the gradients of the cost function are nullified as the system size becomes larger, thus rendering the optimization process almost impossible [23, 24]. Besides, the quantum circuit depth is another issue that constrains this method. The numerous uses of quantum gates, especially CNOT gates, introduce noise and errors, further reducing accuracy [25]. For instance, applying VQE-HEA to molecular systems typically involves numerous CNOT gates, leading to deep circuits that are difficult to optimize for the classical component on current noisy intermediate-scale quantum (NISQ) devices[20].

Advances in Variational Quantum Eigensolvers (VQE) have introduced several innovative methods to improve accuracy and mitigate limitations in NISQ devices. One notable approach is Symmetry-Adapted VQE (SA-VQE), which enforces physically meaningful, symmetry-preserving states. By retaining quantum characteristics such as spin and particle number, SA-VQE ensures more reliable results, particularly for molecular systems [26].

Another improvement, Orbital-Optimized UCC (OO-UCC), decreases the number of variational parameters by using orbital rotations to optimize the Hartree-Fock initial state [27]. State-Averaged Orbital-Optimized VQE (SA-OO-VQE) further improves on this approach by off-loading some of the computations onto classical processors, thus also reducing the sensitivity to noise and qubit limitations [28]. This avoids numerical instabilities near avoided crossings or conical intersections for cases involving degenerate or quasi-degenerate states.

To reduce the impact of hardware noise, Noise-Robust Variational Quantum Eigensolver (NR-VQE) employs techniques such as zero-noise extrapolation and probabilistic error cancellation to increase the reliability of calculations on Noisy Intermediate-Scale Quantum (NISQ) devices [29, 30]. Similarly, the Low-Depth Circuit Ansatz (LDCA) uses sparse parameterized layers with low-density entangling gates to efficiently tackle quantum chemistry problems on noisy hardware [31].

Lastly, QSE-VQE enhances the flexibility of the trial wavefunctions by embedding a higher-dimensional Hilbert space, while reducing the number of CNOT gates. This method achieves greater accuracy with shallower circuits, making it a powerful tool for tackling challenging quantum chemistry problems [32, 33].

1.4 Research Objectives

Regardless of the development of VQE extensions, the problems of circuit depth, symmetry errors, optimization performance, and noise resistance remain, making it impossible to balance the linked

issues fully.

This research will investigate and attempt to solve the HEA-VQE quantum algorithm's problems with large circuit depth, symmetry-breaking, and poor optimization performance. This study aims to use the advantages of HEA and the symmetry-preserving technique to solve many-body issues. We suggest a novel alternative approach, which:

- Utilizes HEA to ensure shallow circuits that can be implemented on near-term quantum hardware.
- Preserves physical symmetries of the Hamiltonian by splitting the diagonalization process into Householder reflection [34, 35] and block diagonalized operations, resulting in accurate and physically meaningful results.
- Minimizes the CNOT gate count while maintaining high accuracy in ground-state energy estimation.

We aim to apply this method to the ground-state energy calculations of several small molecules, such as H_2, LiH, HeH^+, H_3^-, and H_4. The research will center on the comparison of the impact of the proposed method with conventional HEA-VQE, and UCC-VQE in terms of circuit depth, CNOT gate count, and accuracy.

1.5 Outline of the Thesis

I would now like to provide an overview of the work I have completed:

- **Chapter 2: Literature Review** - The write-up will initially cover a survey of existing quantum algorithms for many-body problems, with a special focus on the VQE family and its subvariants.
- **Chapter 3: Theory and Methodology** - The theoretical background and the mathematical model of the proposed quantum algorithms will be presented along with the implementation details.
- **Chapter 4: Results and Discussion** - The computation of numerical results by running the algorithms on test molecules, as well as a discussion on their performance.
- **Chapter 5: Conclusion and Future Work** - A summary of the findings and suggestions for future research directions.

2 Literature Review

2.1 Introduction

Estimation of the ground-state energy is one of the significant tasks in quantum chemistry and physics. Among the classical computational methods, Full Configuration Interaction (FCI) yields excellent results; however, the numerical cost scales exponentially with the number of particles. The traditional (projected) Coupled Cluster Theory (CC) is usually truncated to single, double, or triple excitations and is not exponentially costly. However, it will fail for strongly correlated systems. These limitations make them impractical for large systems (cite Bartlett 2007, Helgaker 2014). Quantum algorithms offer a more efficient alternative to classical methods, particularly those suitable for NISQ devices. This chapter presents a few quantum algorithms for ground-state energy estimations, emphasizing their derivations, merits, and demerits.

2.2 Quantum Algorithms for Ground-State Estimation

2.2.1 Quantum Phase Estimation (QPE)

Quantum Phase Estimation QPE is one of the foundational quantum algorithms to solve the eigenvalue problems of a unitary operator. The idea behind this approach is that if a unitary operator $U = e^{-iH\tau}$ takes action on a quantum state $|\psi\rangle$, then in the case when $|\psi\rangle$ is an eigenvector of the Hamiltonian H, there will be an induced phase ϕ which can be obtained by the Quantum Fourier Transform (QFT):

$$U|\psi\rangle = e^{i\theta}|\psi\rangle, \quad \theta = \frac{E\tau}{\hbar}. \tag{2.1}$$

This phase θ is related to an eigenvalue E which, in this case, is a ground-state energy. QPE gives accurate results but at the cost of requiring deep quantum circuits with large numbers of CNOT gates [15, 36, 37].

Limitations: high CNOT gate count due to QFT; demands extended coherence durations and intricate circuits, rendering it inappropriate for NISQ devices.

2.2.2 Variational Quantum Eigensolver (VQE)

VQE is a hybrid quantum-classical algorithm that aims to find an approximation for the ground-state energy of a given Hamiltonian. This is achieved by minimizing the energy expectation of a parameterized trial wavefunction, which is prepared on a quantum computer. Such an energy function is given by:

$$E(\boldsymbol{\theta}) = \langle\psi(\boldsymbol{\theta})|H|\psi(\boldsymbol{\theta})\rangle, \tag{2.2}$$

where $\boldsymbol{\theta}$ are the variational parameters optimized by a classical algorithm [1].

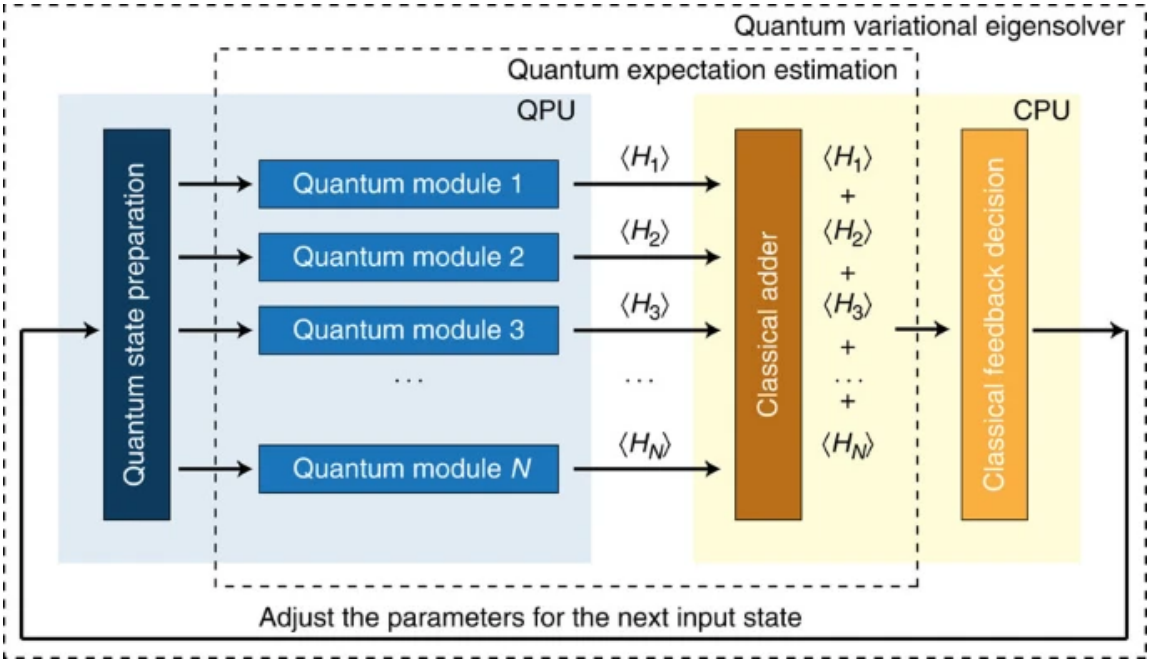


Figure 1: Hybrid quantum-classical architecture for the Quantum Variational Eigensolver (VQE). The quantum processing unit (QPU) calculates expectation values for different quantum modules, which are combined classically to provide feedback to adjust parameters in an iterative loop for finding the ground state energy [1].

2.2.2.1 Hardware-Efficient Ansatz (HEA-VQE)

By construction, the HEA is tailored for current NISQ devices. It consists of a series of interleaved layers of parameterized single-qubit rotations and entangling gates sometimes CNOT gates [17].

The general form of the HEA is:

$$|\psi(\boldsymbol{\theta})\rangle = \prod_{q=1}^{N} \left[U_{\text{rot}}^{q,d}(\theta_{\text{rot}})\right] \times U_{\text{ent}} \times \prod_{q=1}^{N} \left[U_{\text{rot}}^{q,d-1}(\theta_{\text{rot}})\right] \times \ldots \times U_{\text{ent}} \times \prod_{q=1}^{N} \left[U_{\text{rot}}^{q,0}(\theta_{\text{rot}})\right] |\psi_0\rangle, \quad (2.3)$$

where $U_{\text{rot}}(\theta_{\text{rot}})$ represents a layer of a combination of single-qubit rotation gates (e.g., $R_x(\theta)$, $R_y(\theta)$ or $R_z(\theta)$ gates), U_{ent} represents a layer of entangling CNOT gates and d is the depth level of the circuit.

HEA-VQE has been successfully applied to small molecules like H_2, LiH, and BeH_2 [18, 17]. This ansatz is hardware-compatible but has limitations, which include the barren plateaus phenomenon on the optimization landscape [23] and the symmetry-breaking nature that leads to incorrect results with chemical accuracy [22, 20].

2.2.2.2 Symmetry-Preserving Ansatzes

The Unitary Coupled-Cluster Ansatz is one of the most used symmetry-preserving ansätze in quantum chemistry. Theoretically, UCC ansatz is generally considered to have properties of preserving basic symmetries such as particle number and spin [19, 20]. The basis of the UCC ansatz, in turn, lies in classical coupled-cluster theory, where this cluster operator enables the excitations from a reference usually taken in the Hartree-Fock state.

The general form of the UCC ansatz is:

$$|\psi_{\mathrm{UCC}}\rangle = e^{T-T^\dagger}|\psi_{\mathrm{HF}}\rangle, \tag{2.4}$$

where $|\psi_{\mathrm{HF}}\rangle$ represents the Hartree-Fock reference state, and T denotes the cluster operator, which is conventionally limited to single and double excitations (UCCSD):

$$T = \sum_{ia} t_i^a a_a^\dagger a_i + \sum_{ijab} t_{ij}^{ab} a_a^\dagger a_b^\dagger a_i a_j, \tag{2.5}$$

The UCC ansatz is symmetric; it has significant circuit depth due to the exponential number of terms within the cluster operator, especially when dealing with larger systems. In order to realize these terms as quantum circuits, trotterization has to be used, which yields a large number of entangling gates involved, especially CNOT gates [19].

2.2.2.3 ADAPT-VQE

The Adaptive Derivative-Assembled Pseudo-Trotter (ADAPT-VQE) method was proposed to improve the UCC ansatz by mitigating issues due to rapid circuit depth growth and excessive resource consumption. Instead of using a fixed ansatz, ADAPT-VQE constructs it iteratively by picking the most relevant operators from a predefined pool based on the gradient of energy at that iteration [21].

Below is the form of the ansatz in ADAPT-VQE:

$$|\psi(\theta)\rangle = e^{\theta_1 A_1} e^{\theta_2 A_2} \cdots e^{\theta_n A_n} |\psi_0\rangle, \tag{2.6}$$

Here, A_i denotes the chosen operators, which are typically either Pauli operators or fermionic excitation operators, while θ_i represents the variational parameters.

1. Operator Pool: ADAPT-VQE chooses operators from a predetermined set, which includes single and double excitations:

$$A = \{a_i^\dagger a_j, a_i^\dagger a_j^\dagger a_k a_l, \dots\}, \tag{2.7}$$

2. Gradient-Based Selection: In every cycle, the operator that introduces the biggest energy gradient is chosen:

$$\frac{\partial E(\theta)}{\partial \theta_i} = \langle \psi(\theta)|[A_i, H]|\psi(\theta)\rangle, \tag{2.8}$$

3. Dynamic Ansatz Growth: The ansatz is developed in an iterative manner, guaranteeing that only the most relevant terms are incorporated, which leads to a more streamlined circuit with a reduced number of CNOT gates. Through the dynamic construction of the ansatz, ADAPT-VQE minimizes the inclusion of superfluous parameters and gates.

ADAPT-VQE provides two main advantages:

- Circuit Efficiency: By selecting only the most important terms, ADAPT-VQE reduces the number of CNOT gates needed, making it more feasible on NISQ devices.
- Improved accuracy: Since the ansatz is engineered to minimize energy iteratively, ADAPT-VQE often has better accuracy compared to static ansatz methods like HEA.

The number of CNOT gates used in ADAPT-VQE is usually much smaller than that of UCC because the construction of the ansatz is iterative and, therefore, only high-importance excitations are included.

2.2.2.4 Orbital-Optimized UCC (OO-UCC)

The OO-VQE improves VQE by optimizing the molecular orbitals in concert with the ansatz parameters. This decreases the number of parameters and provides a better initial guess of the Hartree-Fock state. For this, the wavefunction takes the form of:

$$|\psi_{OO-UCC}\rangle = e^{T(\theta)-T^{\dagger}(\theta)} R(\kappa)|\psi_{HF}\rangle, \tag{2.9}$$

In this context, $R(\kappa)$ denotes the operator responsible for orbital rotation. Although this approach facilitates faster convergence and yields more precise outcomes, it incurs an increased level of classical computational complexity [27].

2.2.2.5 Noise-Robust VQE (NR-VQE)

Noise-Robust Variational Quantum Eigensolver NR-VQE embeds different methods of error mitigation, like zero-noise extrapolation and probabilistic error cancellation, to reduce hardware noise effects [29, 30]. The energy estimation now has to take into account the presence of noise:

$$E_{\text{NR-VQE}} = E(\boldsymbol{\theta}) + \sum_{\text{errors}} c_i \delta E_i, \tag{2.10}$$

where δE_i are error terms due to the noise. However, Error mitigation introduces additional measurements and computational overhead. Scalability also remains a challenge for larger systems.

2.2.3 Quantum Subspace Expansion (QSE)

Methods of subspace expansion allow for approximations to the ground-state and excited states energies by enlarging the variational search space. The method consists of the diagonalization of the Hamiltonian in a non-orthogonal basis of many-body states [38]. Mathematically, the generalized eigenvalue problem reads:

$$Hc = ScE, \tag{2.11}$$

where H represents the Hamiltonian matrix, S is the overlap matrix, c is the coefficients of the eigenstates in the selected basis, and E are the eigenvalues.

The matrix elements of S and H are calculated by

$$S_{ij} = \langle \Phi_i | \Phi_j \rangle, \tag{2.12}$$
$$H_{ij} = \langle \Phi_i | H | \Phi_j \rangle, \tag{2.13}$$
$$\tag{2.14}$$

in which Φ_i are the many-body basis states.

A variety of methods are used to build the many-body basis for QSE simulations:

- **Truncated configuration interaction expansion:** This method was introduced by McClean et al. The Hamiltonian is diagonalized in the span of states $a_i^\dagger a_j |\phi\rangle$, where $|\phi\rangle$ is the reference state recovered from VQE's run [39].
- **Krylov subspace construction:** The Quantum Lanczos (QLanczos) algorithm establishes the basis through the imaginary-time evolution of a reference state that is sampled at consistent intervals within imaginary time. This results in the formation of a classical Krylov space, thereby enhancing both accuracy and efficiency [40].

Despite these advantages, QSE does face a few challenges, which are as follows:

- Linearly dependent bases, which introduce numerical instabilities in solving the eigenvalues.
- Extra resources to compute the off-diagonal matrix elements, especially for real-time propagation techniques [41].

2.3 Conclusion

This chapter has reviewed some of the major quantum algorithms used for ground-state energy estimation, namely, QPE, VQE, and its extensions like HEA-VQE, UCC-VQE, and ADAPT-VQE. We have discussed the mathematical background of such methods, strengths, and weaknesses while focusing on the specific task of reducing the number of CNOT gates and enhancing their robustness for NISQ devices.

3 Methodology

3.1 Overview of the Proposed Method

In this work, we introduce the Variational Hyperspace Projection (VHyP) approach, which unlocks the use of HEA without the problem of large circuit depth, symmetry-breaking, and poor optimization landscape. The VHyP algorithm splits the diagonalization process into two steps; to block-diagonalize the Hamiltonian in a symmetry-conserved subspace using a Householder reflection transformation [34, 35]; to find the eigenvalues of reduced block Hamiltonian using a unitary transformation. The ground state can then be decomposed into:

$$|\Psi_{\mathrm{VHyP}}\rangle = \omega|\Psi_{\mathrm{HF}}\rangle + \nu\sqrt{(1-\omega^2)}R(\theta)|\Phi_1\rangle, \tag{3.1}$$

where $\nu = \pm 1$ is a relative phase factor and the $\omega = \langle\psi_{\mathrm{HF}}|\psi_{\mathrm{VHyP}}\rangle$ is a projection of the Trial wavefunction $|\psi_{\mathrm{VHyP}}\rangle$ on the Hartree-Fock (HF) states in the physical space of the Hilbert space.

Then, the ground-state energy is approximated with

$$\mathbf{H}^R = R(\theta)^\dagger \mathbf{H} R(\theta), \tag{3.2}$$

$$E(\boldsymbol{\theta}) = \langle\Psi_{\mathrm{VHyP}}|\mathbf{H}^R|\Psi_{\mathrm{VHyP}}\rangle, \tag{3.3}$$

In this context, $R(\theta)$ represents the variational Householder reflection operator, and $\mathbf{H}^R$ is the reduced block-diagonal Hamiltonian that isolated the ground state contribution. This method decreases the number of CNOT gates needed and preserves the symmetry with high accuracy.

3.2 Theory

The key to this strategy is to split the whole process of the Hamiltonian diagonalization into two well-separated subprocesses:

- Computing the Householder reflection $\mathbf{R}$, which block diagonalizes the Hamiltonian, and
- Determining the unitary transformation $\mathbf{P}$ that diagonalizes the block Hamiltonian in the Householder basis.

The two-step decomposition commonly adopts the so-called PR decomposition. The expression $\mathbf{U} = \mathbf{PR}$ enables transformation between the computational basis and the eigenstate basis. In this way, such a strategy can reduce the system dimension significantly while preserving properties of the Hamiltonian, such as symmetry, which is important for speeding up the quantum simulation process.

3.2.1 Variational Principle

Let us consider a $N^F \times N^F$ electronic Hamiltonian operator $\hat{\mathbf{H}}$ of a system containing N^o orbitals such that $N^F = 2^{N^o}$, and where the ground state is nondegenerate. The Hamiltonian is represented in terms of the many-body Fock-space basis set $\{|\Phi_i\rangle\}$ as

$$\hat{\mathbf{H}} = \sum_{ij} H_{ij} |\Phi_i\rangle\langle\Phi_j| \tag{3.4}$$

and $\mathbf{H}$ the Hamiltonian matrix containing the $N^F \times N^F$ elements H_{ij}. In the eigenvector basis set, $\hat{\overline{\mathbf{H}}} = \mathbf{P}^\dagger \hat{\mathbf{H}} \mathbf{P}$,

$$\hat{\overline{\mathbf{H}}} = \sum_{i} E_i |\Psi_i\rangle\langle\Psi_i|. \tag{3.5}$$

with $|\Psi_i\rangle = \mathbf{P}^\dagger |\Phi_i\rangle$ and the Hamiltonian matrix in the eigenstate representation $\overline{\mathbf{H}} = \mathbf{P}^\dagger \mathbf{H} \mathbf{P}$ is diagonal $\overline{H}_{ij} = E_i \delta_{ij}$ with E_i corresponding the the i^{th} eigenvalue sorted in ascending order.
The associated density matrix operators read in the computational and eigenvector basis set as

$$\hat{\mathbf{\Gamma}} = \sum_{ij} |\Phi_i\rangle\langle\Phi_j|; \quad \hat{\overline{\mathbf{\Gamma}}} = \sum_{ij} |\Psi_i\rangle\langle\Psi_j| \tag{3.6}$$

Let now consider the N-body ground state density matrix $\overline{\mathbf{\Gamma}}$ and $\mathbf{\Gamma}$ defined as

$$\overline{\mathbf{\Gamma}} = \langle\Psi_0|\hat{\overline{\mathbf{\Gamma}}}|\Psi_0\rangle; \quad \mathbf{\Gamma} = \langle\Psi_0|\hat{\mathbf{\Gamma}}|\Psi_0\rangle \tag{3.7}$$

where $|\Psi_0\rangle$ denote the ground state vector.

The ground state energy E_0 is computed as the convolution

$$E_0 = tr.\left(\mathbf{\Gamma}\mathbf{H}\right) = tr.\left(\overline{\mathbf{\Gamma}}\overline{\mathbf{H}}\right) = \overline{H}_{00} \tag{3.8}$$

$\overline{\mathbf{H}}$ being diagonal with eigenvalue sorted in ascending order. Assuming the form of the ground state density matrix $\overline{\mathbf{\Gamma}}$ is known to be $\overline{\Gamma}_{kl} = \delta_{k0}\delta_{l0}$, it can be shown (see Appendix 7.1) that for all parametrized unitary transformation $\mathbf{P}(\boldsymbol{\theta})$

$$E(\boldsymbol{\theta}) = tr.\left(\overline{\mathbf{\Gamma}}\mathbf{P}^\dagger(\boldsymbol{\theta})\mathbf{H}\mathbf{P}(\boldsymbol{\theta})\right) \geq E_0 \tag{3.9}$$

with $E(\boldsymbol{\theta}) = E_0$ for $\mathbf{P}(\boldsymbol{\theta}) = \mathbf{P}$.

The variational principle (3.9) can be rewritten as:

$$E(\boldsymbol{\theta}) = \langle\Phi_0|\mathbf{P}^\dagger(\boldsymbol{\theta})\hat{\mathbf{H}}\mathbf{P}(\boldsymbol{\theta})|\Phi_0\rangle \tag{3.10}$$

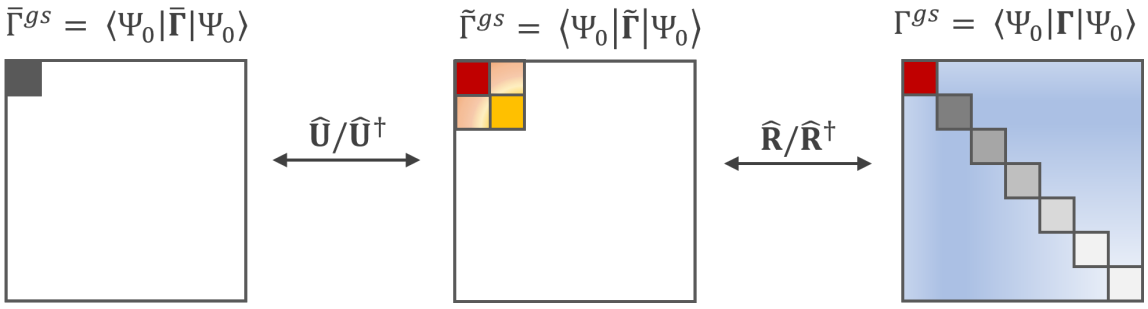


Figure 2: Schematic representation of the ground-state many-body density matrices $\overline{\mathbf{\Gamma}}$, $\widetilde{\mathbf{\Gamma}}$, and $\mathbf{\Gamma}$ in the three different representations, namely the eigenvector, the Householder and the many-body basis set representation. The unitary transformations $\mathbf{U}$, $\mathbf{R}$, and $\mathbf{P}$ linking the three representations are displayed.

3.2.2 Householder Reflection Representation

The VHyP approach introduces an intermediate representation that uses a Householder reflection transformation to block-diagonalize the density matrix, simplifying the computation of E_0. Considering $\mathbf{\Gamma}$ as an idempotent matrix, We define an intermediate density matrix as $\widetilde{\mathbf{\Gamma}}$:

$$\widetilde{\mathbf{\Gamma}} = \mathbf{R}\mathbf{\Gamma}\mathbf{R} = \begin{bmatrix} \widetilde{\mathbf{\Gamma}}^R & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \tag{3.11}$$

where $\widetilde{\mathbf{\Gamma}}^R$ is a 2×2 submatrix given by:

$$\widetilde{\mathbf{\Gamma}}^R = \begin{bmatrix} \omega^2 & \sqrt{\omega^2(1-\omega^2)} \\ \sqrt{\omega^2(1-\omega^2)} & 1-\omega^2 \end{bmatrix}. \tag{3.12}$$

The parameter ω^2 represents the weight of the state $|\Phi_0\rangle$ in the ground state $|\Psi_0\rangle$. Using a Householder reflection $\mathbf{R}$, where $\mathbf{R} = \mathbf{R}^\dagger$ and $|\Phi_0\rangle = \mathbf{R}|\Phi_0\rangle$. Note that, $|\Phi_0\rangle$ is preserved under the $\mathbf{R}$ operation, and if the system's symmetry were encoded in $|\Phi_0\rangle$ then the symmetry would be conserved.

On this note, the ground-state energy can be rewritten as:

$$E_0 = \mathrm{tr}\left(\widetilde{\mathbf{\Gamma}}\widetilde{\mathbf{H}}\right) = \mathrm{tr}\left(\widetilde{\mathbf{\Gamma}}^R\widetilde{\mathbf{H}}^R\right), \tag{3.13}$$

where $\widetilde{\mathbf{H}} = \mathbf{R}\mathbf{H}\mathbf{R}$ and

$$\widetilde{\mathbf{H}}^R = \begin{bmatrix} H_{00} & \widetilde{H}_{10} \\ \widetilde{H}_{01} & \widetilde{H}_{11} \end{bmatrix}. \tag{3.14}$$

To simplify the optimization, we reduce the variational problem by introducing a variational Householder reflection transformation $\mathbf{R}(\boldsymbol{\theta})$ and the seed of symmetry into the variational principle as:

$$E_0 = \min_{\omega,\boldsymbol{\theta}} \left[\mathrm{tr}\left(\widetilde{\mathbf{\Gamma}}^R(\omega)\mathbf{H}^R(\boldsymbol{\theta})\right)\right], \tag{3.15}$$

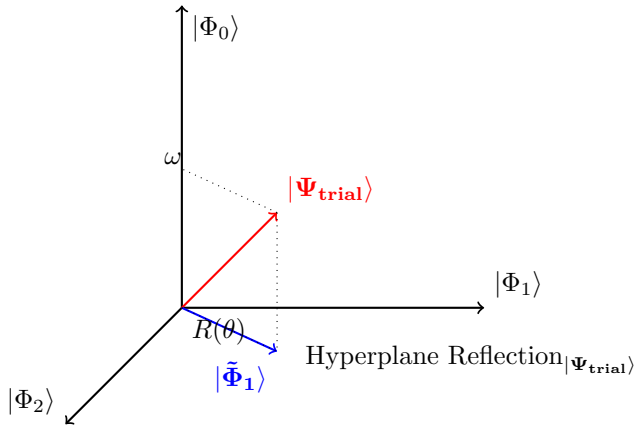
where $\mathbf{H}(\boldsymbol{\theta}) = \mathbf{R}(\boldsymbol{\theta})\mathbf{H}\mathbf{R}(\boldsymbol{\theta})$ and

$$\mathbf{H}^R(\boldsymbol{\theta}) = \begin{bmatrix} H_{00} & H_{10}(\boldsymbol{\theta}) \\ H_{01}(\boldsymbol{\theta}) & H_{11}(\boldsymbol{\theta}) \end{bmatrix}. \tag{3.16}$$

This approach has efficiently reduced the complexity of the full Hilbert space $N^F \times N^F$ to a 2×2 subspace Hamiltonian, which conserved the symmetry of the $|\Phi_0\rangle$.

3.2.3 Ground state wavefunction decomposition and Hyperspace Projection

To be consistent with the reduced dimensionality of the Hilbert space, we propose a decomposition of a trial wavefunction using the geometrically intuitive perspective. The ground state vector is decomposed into contributions along $|\Phi_0\rangle$ and its orthogonal component $|\widetilde{\Phi}_1\rangle$ as shown in the figure below:



and this follows the mathematical expression:

$$|\psi_{\text{trial}}\rangle = \omega|\Phi_0\rangle + \nu\sqrt{(1-\omega^2)}|\widetilde{\Phi}_1\rangle, \tag{3.17}$$

where $\nu = \pm 1$ is a relative phase factor, $\omega = \langle\Phi_0|\psi_{\text{trial}}\rangle$ is a projection of the trial wavefunction $|\psi_{\text{trail}}\rangle$, and $|\widetilde{\Phi}_1\rangle = \mathbf{R}(\boldsymbol{\theta})|\Phi_1\rangle$. In this case, the variational principle defined in Eq. (3.15) involves searching for the complement vector $|\widetilde{\Phi}_1(\boldsymbol{\theta})\rangle = \mathbf{R}(\boldsymbol{\theta})|\Phi_1\rangle$ to the initial vector $|\Phi_0\rangle$, which together form the complete ground state. Assuming $|\Phi_0\rangle$ to be a good approximation to the true ground state, i.e., $|\Phi_0\rangle$ has some weight $\omega^2 = |\langle\Phi_0|\Psi_0\rangle|^2 > 0.5$.

3.3 Symmetry-Preserved Cost Functions

To define the efficient cost functions suitable for this approach for practical implementation, we guess that $|\Phi_0\rangle$ is a Hartree-Fock (HF) state in the physical space of the Hilbert space. Since the Hartree-Fork mean-field solution can be computed using a quantum chemistry application, then its state vector can be presented as a pure state on a quantum circuit. This is a logical reason to start the variational process from the HF solution, and we introduce two alternate cost functions that lower the energy landscape and preserve symmetries. Likewise, these cost functions allow for

subprocesses or iterative use of quantum circuits, which systematically reduce the depth of the quantum circuit to be used.

3.3.1 Energy Gain Cost Function

The first proposed cost function is the *Energy Gain* function, denoted by $E_G(\boldsymbol{\theta})$. It quantifies the energy stabilization arising from the interaction between $|\Phi_0\rangle$ and $|\widetilde{\Phi}_1(\boldsymbol{\theta})\rangle$:

$$E_G(\boldsymbol{\theta}) = \frac{\Delta(\boldsymbol{\theta})}{2}\left[1 - \sqrt{1 + \frac{4\widetilde{H}_{01}(\boldsymbol{\theta})\widetilde{H}_{10}(\boldsymbol{\theta})}{\Delta(\boldsymbol{\theta})^2}}\right], \tag{3.18}$$

where $\Delta(\boldsymbol{\theta}) = |\widetilde{H}_{11}(\boldsymbol{\theta}) - H_{00}|$ is the energy difference between the states determined by $|\Phi_0\rangle$ and $|\widetilde{\Phi}_1(\boldsymbol{\theta})\rangle$. As shown in Appendix 7.1.1, this function does not diverge if $\Delta(\boldsymbol{\theta}) = 0$ as long as $|\Phi_0\rangle$ and $|\widetilde{\Phi}_1(\boldsymbol{\theta})\rangle$ are orthogonal.

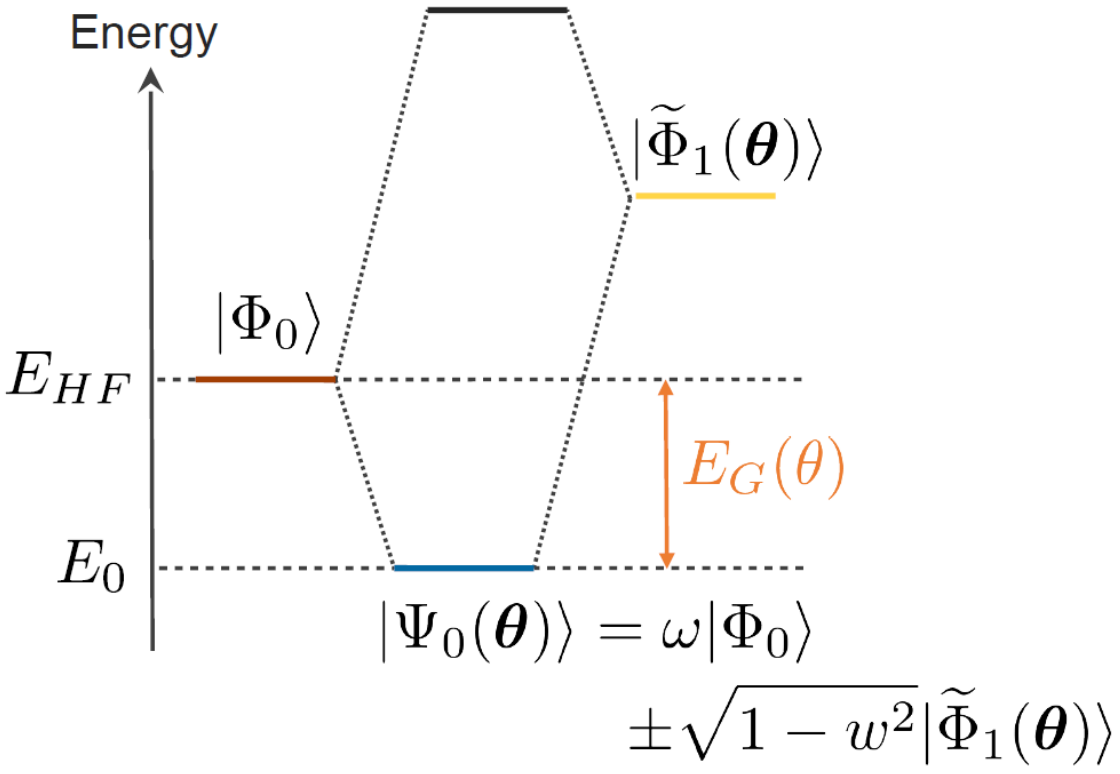


Figure 3: Schematic representation of the "2 level" decomposition of the ground state. The strategy we propose consist in optimizing the gain energy E_G issued from the interaction between the *good-guess* $|\Phi_0\rangle$ and a variational vector $|\widetilde{\Phi}_1(\boldsymbol{\theta})\rangle$.

Notably, $E_G(\boldsymbol{\theta})$ vanishes when the trial states $|\Phi_0\rangle$ and $|\widetilde{\Phi}_1(\boldsymbol{\theta})\rangle$ do not belong to the same symmetry subspace of the Hamiltonian: $H_{01}(\boldsymbol{\theta}) = \langle\Phi_0|\hat{\mathbf{H}}|\widetilde{\Phi}_1(\boldsymbol{\theta})\rangle = 0$. This symmetry preserving feature simplifies the energy landscape, while ensuring that at optimal parameters $\boldsymbol{\theta}^*$, $|\widetilde{\Phi}_1(\boldsymbol{\theta}^*)\rangle$ has the same quantum numbers - e.g. particle number $\langle N\rangle$ and spin $\langle S^2\rangle$, - as $|\Phi_0\rangle$. Thus, taking $E_G(\boldsymbol{\theta})$ as the objective function renders HEA in VQE algorithms much more straightforward to implement while retaining physical consistency. Now, the ground state energy can be evaluated as:

$$E(\boldsymbol{\theta}^*) = H_{00} + E_G(\boldsymbol{\theta}^*) \tag{3.19}$$

The subprocess of this cost function is shown below:

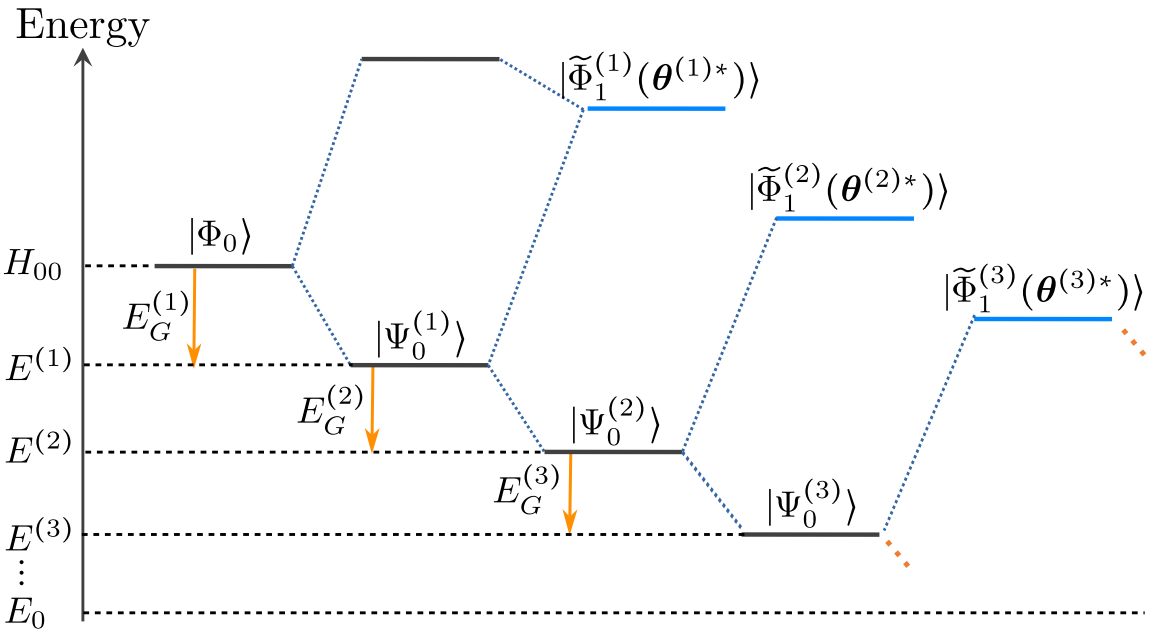


Figure 4: Schematic depiction of the iterative algorithm for energy gain cost function.

3.3.2 Quantum QR (Tridiagonalization) Cost Function

The second proposed cost function, $\widetilde{H}_{01}(\boldsymbol{\theta})\widetilde{H}_{10}(\boldsymbol{\theta})$, maximizes the product of off-diagonal Hamiltonian terms which implies a QR-based tridiagonalization of the Hamiltonian:

$$C_{\mathrm{QR}}(\boldsymbol{\theta}) = \widetilde{H}_{01}(\boldsymbol{\theta})\widetilde{H}_{10}(\boldsymbol{\theta}). \tag{3.20}$$

As shown in Appendix 7.1.3, maximizing this expression with respect to $\boldsymbol{\theta}$ under the constraint $\mathbf{R}(\boldsymbol{\theta})|\Phi_0\rangle = |\Phi_0\rangle$ results in a tridiagonal Hamiltonian where $\widetilde{H}_{0j}(\boldsymbol{\theta}) = 0$ for $j > 1$. The ground state energy for this cost function is the lowest eigenvalue of the 2×2 reduced Hamiltonian.

This tridiagonalization permits the ground state to be expressed using only the major components along $|\Phi_0\rangle$ and $|\widetilde{\Phi}_1(\boldsymbol{\theta})\rangle$, greatly reducing computational overhead. Iteratively applying this process results in a full tridiagonal form of the Hamiltonian, similar in spirit to the QR algorithm for matrix diagonalization.

3.4 Quantum Algorithm Framework

In practice, implementing a reflection operator $\mathbf{R}$ on quantum hardware requires multi-controlled NOT gates, which can be resource-intensive [42, 43]. To somewhat alleviate this, we propose an implementation of a unitary approximation of $\mathbf{R}$ with a hardware-efficient ansatz (HEA), since the Householder reflection transformation is a unique unitary transformation among the numerous unitary transformations.

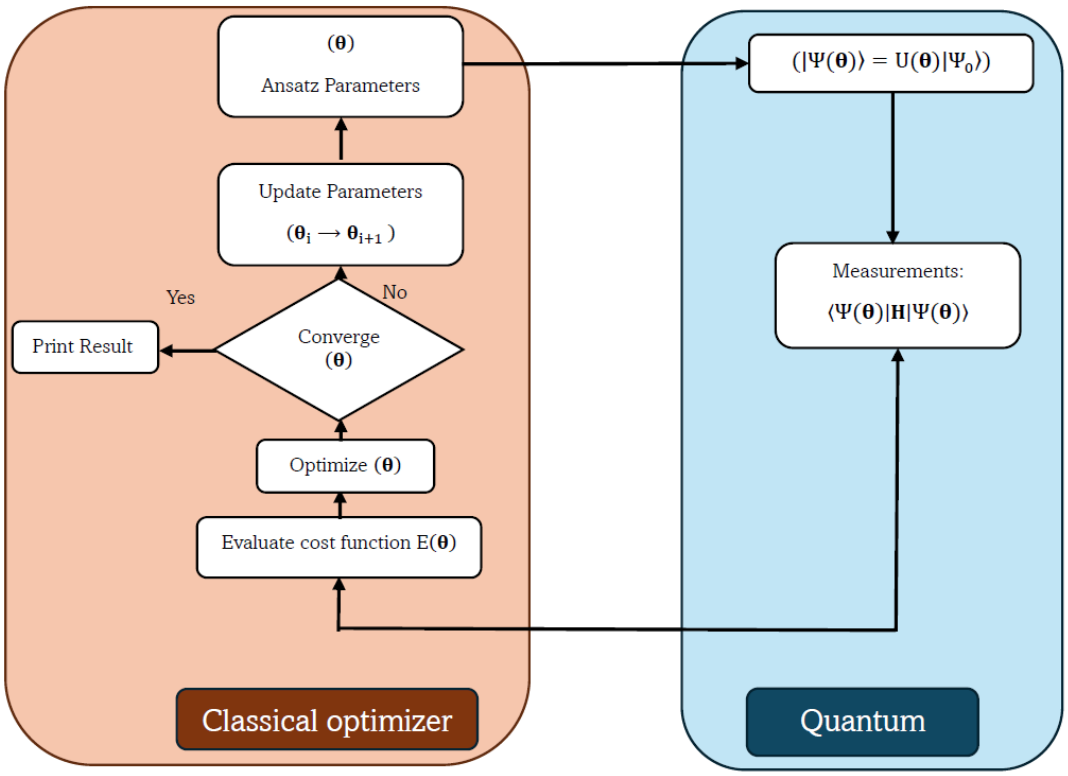


Figure 5: VQE architecture as a hybrid quantum-classical algorithm

In contrast to standard VQE, as shown in figure 5, our approach constructs the ground state progressively without an explicit quantum circuit representation as shown in figure 6. The trial wavefunction can be defined as:

$$|\Psi_{\text{trial}}\rangle = \omega|\Phi_0\rangle \pm \sqrt{1-\omega^2}\,|\widetilde{\Phi}_1\rangle, \tag{3.21}$$

where $|\widetilde{\Phi}_1\rangle = U(\boldsymbol{\theta})|\Phi_1\rangle$ and $U(\boldsymbol{\theta})$ is directly implemented using HEA on quantum computer.

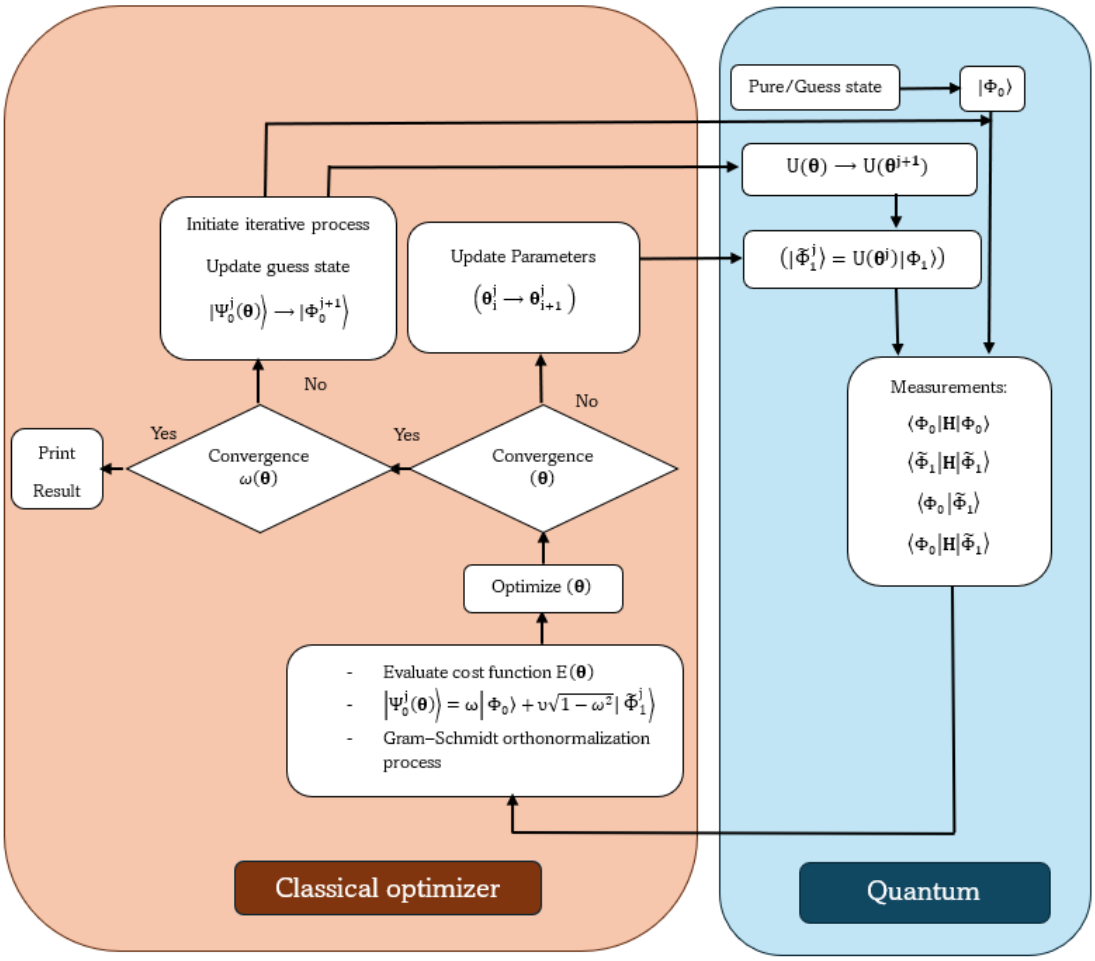
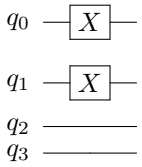


Figure 6: Propose architecture as a hybrid quantum-classical algorithm

Initialization: The algorithm starts by constructing a pure state for selecting the Hartree-Fork state in the physical space of the Hilbert space. HF energy is readily calculated from the pySCF package and is considered H_{00}.

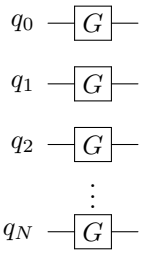
It follows: $|\Psi_{(0)}\rangle = |\Phi_0\rangle$ = HF determinant

For instance, The circuit construction of the HF state of a 4-qubit system in the physical space ($|1100\rangle$) is shown below:



Iteration 1: Initial state $|\Phi_1\rangle$ can act as a state with the same symmetry as HF state or fully Hadamard gates or variational gates. This initial state can be easily constructed on a quantum computer in many different ways:

$|\Phi_1\rangle$:



The gate G can be X-gate, H-gate or RY($\boldsymbol{\theta}$)

The goal is to optimize the set of parameters $\boldsymbol{\theta}^1$ of the full quantum circuit $|\widetilde{\Phi}_1\rangle = U(\boldsymbol{\theta^{(1)}})|\Phi_1\rangle$, where $U(\boldsymbol{\theta^{(1)}})$ is a form of Hardware efficient ansatz (HEA) construct with Ry(θ), $\mathrm{U}_2(\theta, \phi)$, $\mathrm{U}_3(\theta, \phi, \gamma)$ gates and CNOT gate cascade.

The quantum circuit $|\widetilde{\Phi}_1\rangle$ can be constructed on real QC as (for instance 4-qubit system with $depth = 1$ to $depth = n$):

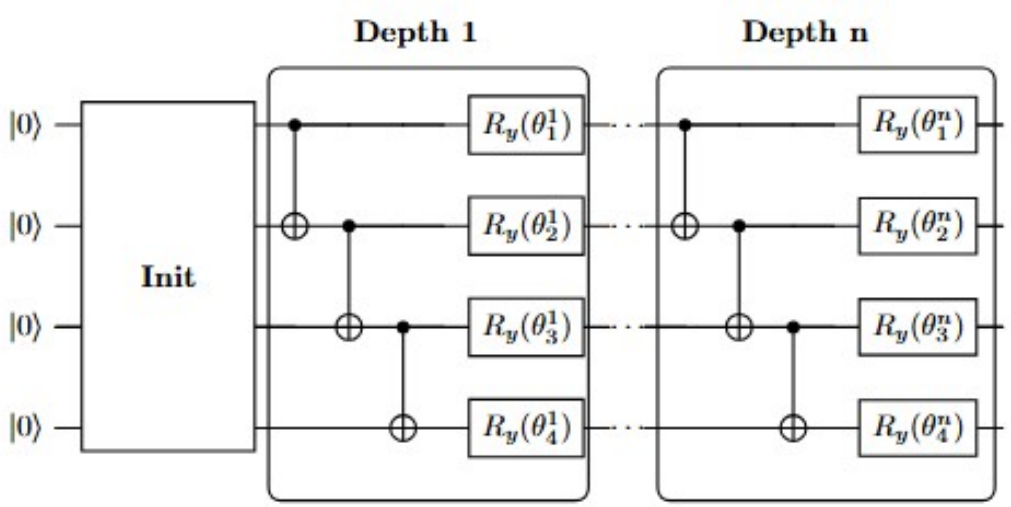


Figure 7: Schematic representation of the HEA using an initialization layer followed by n depths containing variational and entangling gates.

3.4.1 Measurement Consideration

Using $E_G(\boldsymbol{\theta})$ as the cost function requires additional measurements compared to standard VQE, specifically for $\widetilde{H}_{11}(\boldsymbol{\theta})$ and $|\widetilde{H}_{01}(\boldsymbol{\theta})|^2$. The former can be obtained from standard Hamiltonian measurements. The latter corresponds to state fidelity measurements, which are possible without ancilla qubits or Hadamard tests [32, 44]. So that $|\Phi_0\rangle$ is kept fixed during the optimization, a Gram-Schmidt procedure can impose this condition implicitly at the cost of some small additional measurement. Therefore, this ansatz-preserving strategy requires three more measurements than standard VQE techniques.

The measurement is carried out as follows:

1. The overlap:
$$S(\boldsymbol{\theta}^{(1)}) = \langle \Phi_0 | \widetilde{\Phi}_1(\boldsymbol{\theta}^{(1)}) \rangle \tag{3.22}$$

2. The off diagonal reduced Hamiltonian elements:
$$\widetilde{H}_{01}(\boldsymbol{\theta}^{(1)}) = \langle \Phi_0 | \mathbf{H} | \widetilde{\Phi}_1(\boldsymbol{\theta}^{(1)}) \rangle = \widetilde{H}_{10}(\boldsymbol{\theta}^{(1)}) \tag{3.23}$$

3. The anti-HF Hamiltonian element:
$$\widetilde{H}_{11}(\boldsymbol{\theta}^{(1)}) = \langle \widetilde{\Phi}_1(\boldsymbol{\theta}^{(1)}) | \mathbf{H} | \widetilde{\Phi}_1(\boldsymbol{\theta}^{(1)}) \rangle \tag{3.24}$$

The cost functions are to be minimized under the constrain ($\langle \Phi_0 | \widetilde{\Phi}_1(\boldsymbol{\theta}^{(1)}) \rangle = 0$) but with the implementation of Gram–Schmidt orthonormalization process, this has been taking care of to avoid constrain minimization.

- Energy gain Cost function: The variational process by the classical optimizer is done through this cost function:

$$E_G(\boldsymbol{\theta}^{(1)}) = \frac{\Delta(\boldsymbol{\theta}^{(1)})}{2}\left[1 - \sqrt{1 + \frac{4\widetilde{H}_{01}(\boldsymbol{\theta}^{(1)})\widetilde{H}_{10}(\boldsymbol{\theta}^{(1)})}{\Delta(\boldsymbol{\theta}^{(1)})^2}}\right], \tag{3.25}$$

where $\Delta(\boldsymbol{\theta}^{(1)}) = |\widetilde{H}_{11}(\boldsymbol{\theta}^{(1)}) - H_{00}|$, the ground state energy evaluated can be written as:

$$E(\boldsymbol{\theta}^{*(1)}) = H_{00} + E_G((\boldsymbol{\theta}^*)^{(1)}) \tag{3.26}$$

- Quantum QR (Tridiagonalization) Cost Function: This cost function carries out the variational process with classical optimizer by maximizing the off-diagonal element of the reduced Hamiltonian as:

$$C_{\mathrm{QR}}(\boldsymbol{\theta}^{(1)}) = \widetilde{H}_{01}(\boldsymbol{\theta}^{(1)})\widetilde{H}_{10}(\boldsymbol{\theta}^{(1)}) \tag{3.27}$$

The diagonalization of the 2×2 reduced Hamiltonian can be done easily:

$$\widetilde{\mathbf{H}}^{C^1}(\boldsymbol{\theta}^{(1)}) = \begin{bmatrix} H_{00} & \widetilde{H}_{01}(\boldsymbol{\theta}^{(1)}) \\ \widetilde{H}_{10}(\boldsymbol{\theta}^{(1)}) & \widetilde{H}_{11}(\boldsymbol{\theta}^{(1)}) \end{bmatrix}. \tag{3.28}$$

and the lowest eigenvalue corresponds to the ground state energy.

The lowest eigenstate is computed as:

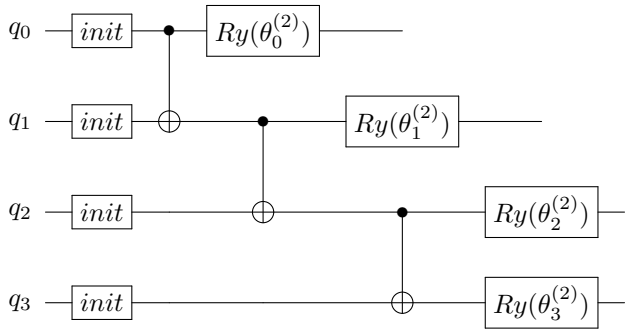
$$|\Psi_0(\boldsymbol{\theta}^{*(1)})\rangle = \omega^{(1)}|\Phi_0\rangle \pm \sqrt{1 - (\omega^{(1)})^2}\,|\widetilde{\Phi}_1(\boldsymbol{\theta}^{*(1)})\rangle, \tag{3.29}$$

where $(\boldsymbol{\theta}^*)^{(1)}$ is the optimized parameter for the first iteration, $\omega^{2(1)}$, the weight of $|\phi^0\rangle$, have to be check to convergence above 0.5. When the optimization results have converged, they will be analyzed and compared to the exact values. This algorithm's subprocess is subsequently implemented through iteration 2 methods.

Iteration 2: For the initialization of the second iteration, the $|\Phi_0\rangle$ is replaced by $|\Psi_0(\boldsymbol{\theta}^{*(1)})\rangle$ and the H_{00} is replaced by $E(\boldsymbol{\theta}^{*(1)})$.

A new set of optimize parameters $\boldsymbol{\theta}^{(\mathbf{2})}$ are provided for the HEA $U(\boldsymbol{\theta}^{(\mathbf{2})})$ and for the second time the quantum circuit $|\widetilde{\Phi}_1^2\rangle = U(\boldsymbol{\theta}^{(\mathbf{2})})|\Phi_1\rangle$ has to be optimized.

The quantum circuit $|\widetilde{\Phi}_1^2\rangle$ is showed below:



The measurements follow by:

1. The overlaps:

$$S_{02} = \langle \Phi_0 | \widetilde{\Phi}_1(\boldsymbol{\theta}^{(2)}) \rangle \tag{3.30}$$

$$S_{12} = \langle \widetilde{\Phi}_1(\boldsymbol{\theta}^{*(1)}) | \widetilde{\Phi}_1(\boldsymbol{\theta}^{(2)}) \rangle \tag{3.31}$$

2. The reduced Hamiltonian elements:

$$\widetilde{H}_{02}(\boldsymbol{\theta}^{(2)}) = \langle \Phi_0 | \mathbf{H} | \widetilde{\Phi}_1(\theta^{(2)}) \rangle = \widetilde{H}_{20}(\boldsymbol{\theta}^{(2)}) \tag{3.32}$$

3. The off-diagonal reduced Hamiltonian elements:

$$\widetilde{H}_{01}(\boldsymbol{\theta}^{(2)}) = \langle \psi^1(\boldsymbol{\theta}^{*(1)}) | \mathbf{H} | \widetilde{\Phi}_1(\boldsymbol{\theta}^{(2)}) \rangle = \widetilde{H}_{10}(\boldsymbol{\theta}^{(2)}) \tag{3.33}$$

4. The anti-HF Hamiltonian element:

$$\widetilde{H}_{11}(\boldsymbol{\theta}^{(2)}) = \langle \widetilde{\Phi}_1(\boldsymbol{\theta}^{(2)}) | \mathbf{H} | \widetilde{\Phi}_1(\boldsymbol{\theta}^{(2)}) \rangle \tag{3.34}$$

The cost functions are to be minimized under the constraints ($S_{12} = S_{12} = 0$). This has been implemented through the Gram–Schmidt orthonormalization process.

- Energy gain Cost function:

$$E_G(\boldsymbol{\theta}^{(2)}) = \frac{\Delta(\boldsymbol{\theta}^{(2)})}{2} \left[1 - \sqrt{1 + \frac{4\widetilde{H}_{01}(\boldsymbol{\theta}^{(2)})\widetilde{H}_{10}(\boldsymbol{\theta}^{(2)})}{\Delta(\boldsymbol{\theta}^{(2)})^2}} \right], \tag{3.35}$$

  where $\Delta(\boldsymbol{\theta}^{(2)}) = |\widetilde{H}_{11}(\boldsymbol{\theta}^{(2)}) - E(\boldsymbol{\theta}^{*(1)})|$

  The ground state energy computation follows:

$$E(\boldsymbol{\theta}^{*(2)}) = E(\boldsymbol{\theta}^{*(1)}) + E_G(\boldsymbol{\theta}^{*(2)}) \tag{3.36}$$

- Quantum QR (Tridiagonalization) Cost Function:

$$C_{\mathrm{QR}}(\boldsymbol{\theta}^{(2)}) = \left[\omega^{*(1)} \widetilde{H}_{02}(\boldsymbol{\theta}^{(2)}) + \sqrt{1 - (\omega^{*(1)})^2}\, \widetilde{H}_{12}(\boldsymbol{\theta}^{(2)} \right]^2, \tag{3.37}$$

  The diagonalization of the 2×2 reduced Hamiltonian can be done easily:

$$\widetilde{\mathbf{H}}^{C^2}(\boldsymbol{\theta}^{(1)}) = \begin{bmatrix} E(\boldsymbol{\theta}^{(1)}) & \widetilde{H}_{01}(\boldsymbol{\theta}^{(2)}) \\ \widetilde{H}_{10}(\boldsymbol{\theta}^{(2)}) & \widetilde{H}_{11}(\boldsymbol{\theta}^{(2)}) \end{bmatrix}. \tag{3.38}$$

  The ground state energy is also computed as the lowest eigenvalue of $\widetilde{\mathbf{H}}^{C^2}(\boldsymbol{\theta}^{(2)})$. This iterative cost function is equivalent to triangulating the Hamiltonian $\mathbf{H}$. For instance, the 2×2 reduced Hamiltonian for the second iteration can be replaced by a 3×3 dimension as:

$$\widetilde{\mathbf{H}}^{C^2}(\boldsymbol{\theta}^{(1)},\boldsymbol{\theta}^{(2)}) = \begin{bmatrix} \langle\Phi_0|\mathbf{H}|\Phi_0\rangle & \langle\Phi_0|\mathbf{H}|\widetilde{\Phi}_1(\boldsymbol{\theta}^{(1)})\rangle & \langle\Phi_0|\mathbf{H}|\widetilde{\Phi}_1(\boldsymbol{\theta}^{(2)})\rangle \\ \langle\widetilde{\Phi}_1(\boldsymbol{\theta}^{(1)})|\mathbf{H}|\Phi_0\rangle & \langle\widetilde{\Phi}_1(\boldsymbol{\theta}^{(1)})|\mathbf{H}|\widetilde{\Phi}_1(\boldsymbol{\theta}^{(1)})\rangle & \langle\widetilde{\Phi}^1(\boldsymbol{\theta}^{(1)})|\mathbf{H}|\widetilde{\Phi}_1(\boldsymbol{\theta}^{(2)})\rangle \\ \langle\widetilde{\Phi}_1(\boldsymbol{\theta}^{(2)})|\mathbf{H}|\Phi_0\rangle & \langle\widetilde{\Phi}_1(\boldsymbol{\theta}^{(2)})|\mathbf{H}|\widetilde{\Phi}_1(\boldsymbol{\theta}^{(1)})\rangle & \langle\widetilde{\Phi}_1(\boldsymbol{\theta}^{(2)})|\mathbf{H}|\widetilde{\Phi}_1(\boldsymbol{\theta}^{(2)})\rangle \end{bmatrix}. \tag{3.39}$$

where $\langle\Phi_0|\mathbf{H}|\widetilde{\Phi}_1(\boldsymbol{\theta}^{(2)})\rangle = 0$ and $\langle\widetilde{\Phi}_1(\boldsymbol{\theta}^{(2)})|\mathbf{H}|\Phi_0)\rangle = 0$.

Hence,

$$\widetilde{\mathbf{H}}^{C^2}(\boldsymbol{\theta}^{(1)},\boldsymbol{\theta}^{(2)}) = \begin{bmatrix} \langle\Phi_0|\mathbf{H}|\Phi_0\rangle & \langle\Phi_0|\mathbf{H}|\widetilde{\Phi}_1(\boldsymbol{\theta}^{(1)})\rangle & 0 \\ \langle\widetilde{\Phi}_1(\boldsymbol{\theta}^{(1)})|\mathbf{H}|\Phi_0\rangle & \langle\widetilde{\Phi}_1(\boldsymbol{\theta}^{(1)})|\mathbf{H}|\widetilde{\Phi}_1(\boldsymbol{\theta}^{(1)})\rangle & \langle\widetilde{\Phi}_1(\boldsymbol{\theta}^{(1)})|\mathbf{H}|\widetilde{\Phi}_1(\boldsymbol{\theta}^{(2)})\rangle \\ 0 & \langle\widetilde{\Phi}_1(\boldsymbol{\theta}^{(2)})|\mathbf{H}|\widetilde{\Phi}_1(\boldsymbol{\theta}^{(1)})\rangle & \langle\widetilde{\Phi}_1(\boldsymbol{\theta}^{(2)})|\mathbf{H}|\widetilde{\Phi}_1(\boldsymbol{\theta}^{(2)})\rangle \end{bmatrix}. \tag{3.40}$$

The ground state computation for the second iteration follows:

$$|\Psi_0(\boldsymbol{\theta}^{*(2)})\rangle = \omega^{(2)}|\Psi_0(\boldsymbol{\theta}^{*(1)})\rangle \pm \sqrt{1-(\omega^{(2)})^2}\,|\widetilde{\Phi}_1(\boldsymbol{\theta}^{(2)})\rangle, \tag{3.41}$$

where $(\boldsymbol{\theta}^*)^{(2)}$ is the optimized parameter for the second iteration, $\omega^{(2)}$, the weight of $|\Psi_0(\boldsymbol{\theta}^{*(1)})\rangle$ also follow the optimization to converge to value above 0.5, otherwise the algorithm move to next iteration.

Iteration 3: For the initialization of the third iteration, the $|\Psi_0(\boldsymbol{\theta}^{*(1)})\rangle$ is replaced by $|\Psi_0(\boldsymbol{\theta}^{*(2)})\rangle$ and the $E(\boldsymbol{\theta}^{*(1)})$ is replaced by $E(\boldsymbol{\theta}^{*(2)})$. The iterative process follows the same procedure continuously until the desired lowest eigenvalue is reached.

For *nth* iterative quantum QR cost function, the Hamiltonian tridiagonalarization can be formed as follows:

$$\widetilde{\mathbf{H}}^{C^n}(\boldsymbol{\theta}^{(1)},\boldsymbol{\theta}^{(2)},...\boldsymbol{\theta}^{(n)}) = \begin{bmatrix} \widetilde{H}_{00} & \widetilde{H}_{01} & 0 & 0 & 0 & \dots & 0 \\ \widetilde{H}_{10} & \widetilde{H}_{11} & \widetilde{H}_{12} & 0 & 0 & \dots & 0 \\ 0 & \widetilde{H}_{21} & \widetilde{H}_{22} & \widetilde{H}_{23} & 0 & \dots & 0 \\ 0 & 0 & \widetilde{H}_{32} & \widetilde{H}_{33} & * & \dots & 0 \\ 0 & 0 & 0 & * & * & \ddots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \widetilde{H}_{(n-1)n} \\ 0 & 0 & 0 & 0 & \dots & \widetilde{H}_{n(n-1)} & \widetilde{H}_{nn} \end{bmatrix} \tag{3.42}$$

3.4.2 Optimization Process

The VHyP algorithm optimizes the parameters of the trial wavefunction so that the expectation value of the energy reaches a minimum, thus approximating the ground-state energy of the electronic Hamiltonian. This optimization step is central to the algorithm, as it is directly connected to the accuracy and efficiency of the parameter tuning for the quality of the ground-state estimation. To

do so, we integrate different optimization algorithms, such as COBYLA [45], SLSQP [46], and L-BFGS-B [47], and then compare their performances to determine a suitable one for our framework.

In each iteration, the optimizers minimize the cost functions by performing multiple passes through the parameters, thereby fine-tuning them in order to reach optimum configurations of the trial wavefunction. This iterative improvement is in line with VHyP's objective of efficiently converging to the ground state of the target Hamiltonian.

3.5 Quantum Simulation Environment: myQLM

The VHyP framework is implemented in the myQLM environment [48], which allows state vector operations over the generated electronic Hamiltonians efficiently. This comes in handy when one wants to directly construct the molecular Hamiltonian from PySCF-generated integrals and transform it into a qubit Hamiltonian. The electronic Hamiltonians used are derived from the STO-3G basis set. Using the PySCF library, the one and two-electron integrals are computed, then the fermionic operators are mapped into spin operators to form a qubit Hamiltonian using the Jordan-Wigner transformation. [49]. A mathematical representation of the Jordan-Wigner transformation reads as:

$$\hat{a}_i^\dagger \rightarrow \left(\prod_{j=0}^{i-1} Z_j \right) \sigma_i^+, \tag{3.43}$$

$$\hat{a}_i \rightarrow \left(\prod_{j=0}^{i-1} Z_j \right) \sigma_i^-, \tag{3.44}$$

where σ_i^+ and σ_i^- are the Pauli raising and lowering operators, respectively, and Z_j are Pauli-Z matrices, which enforce the fermionic anti-commutation relations. This transformation maps the molecular Hamiltonian into a qubit Hamiltonian suitable for quantum computation.

3.5.1 Comparison of HEA Approaches in VQE and VHyP

One of the advantages of the VHyP method is that it can enhance the applicability of HEA on quantum devices by embedding hardware efficiency with symmetry preservation. In this work, we compare the implementation of the conventional HEA approach in VQE with the HEA implementation in VHyP.

3.5.1.1 Generic HEA-VQE Ansatz

The generic HEA ansatz for VQE is realized with alternating layers of parameterized single-qubit rotations and U_{ent}, where the rotation parameters are optimized for each layer:

$$|\Psi(\boldsymbol{\theta})_{HEA-VQE}\rangle = \prod_{\text{layers}} \left[\prod_{\text{qubits}} U(\theta_i) \times U_{ent} \right] \times \prod_{\text{qubits}} U(\theta_i)|\Psi_0\rangle. \tag{3.45}$$

This technique is flexible but does not preserve symmetry, and it may result in large circuit depths where the minimum energy is given as:

$$E(\boldsymbol{\theta}) = min_{(\boldsymbol{\theta})} \langle \psi(\boldsymbol{\theta}) | \mathbf{H} | \psi(\boldsymbol{\theta}) \rangle \tag{3.46}$$

3.5.1.2 HEA-VHyP Ansatz

In the VHyP approach, one relaxes the HEA structure to accommodate the symmetry-preserved hyperplane construction. The HEA ansatz in VHyP:

$$|\Psi(\boldsymbol{\theta})_{HEA-VHyP}\rangle = \prod_{\text{layer(s)}} \left[\prod_{\text{qubits}} U(\theta_i) \times U_{ent} \right] |\Phi_1\rangle, \tag{3.47}$$

where $|\Phi_1\rangle$ is a complementary state which has the same symmetry as $|\Psi_{HF}\rangle$ state while the ground state energy is given as:

$$E(\boldsymbol{\theta}) = H_{00} + E_G(\boldsymbol{\theta}) \tag{3.48}$$

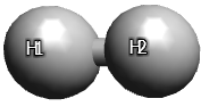
This construction facilitates the orthogonalization implicit in VHyP so that the HF and the complementary states are orthogonal and symmetry-preserving at all times during optimization.

The VHyP ansatz provides a simple, symmetry-aware path to variational optimization, amalgamating the benefits of HEA with higher fidelity through shallower circuits, thus suitable for NISQ devices.

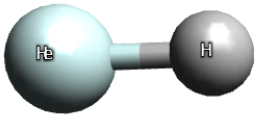
4 Results and Discussion

In this chapter, we present the findings of applying the Variational Hyperspace Projection (VHyP) approach to a variety of quantum chemistry problems. Performance measurements are supplied for each optimizer examined, and comparisons are made with typical HEA-VQE and UCC-VQE procedures to demonstrate VHyP's computational efficiency and accuracy. We tested molecules H_2, HeH^+, LiH, H_3^-, and H_4 in 4-, 6- and 8-qubit systems.

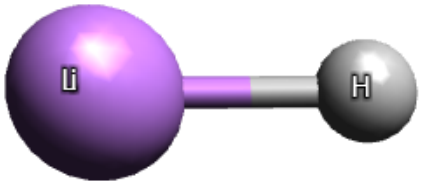
The molecular structure of the systems under testing are shown below:



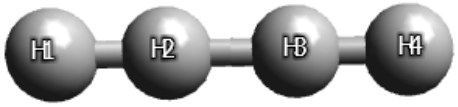
(a) H_2 diagram



(b) HeH^+ diagram



(c) Lithium hydride LiH diagram



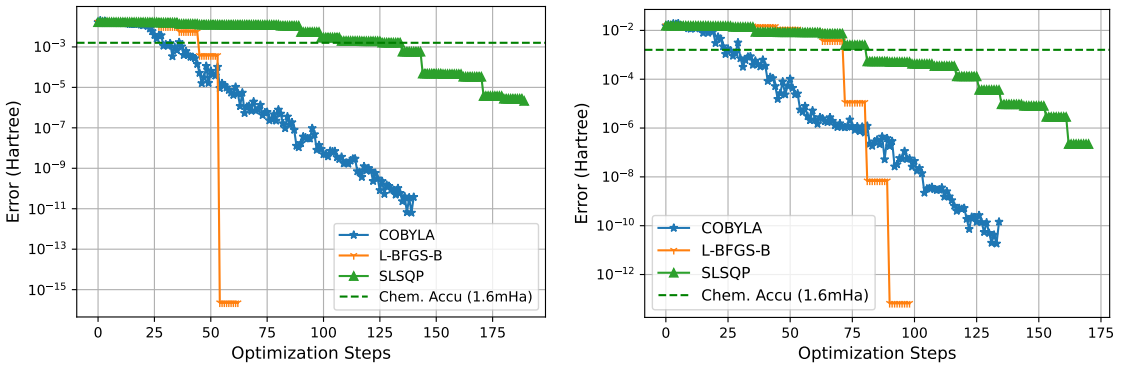
(d) Linear chain of Hydrogen atoms

Figure 8: Molecules under the application of VHyP algorithm

4.0.1 Performance of Optimization Algorithms

The effectiveness of the classical optimization algorithms COBYLA, L-BFGS-B, and SLSQP was evaluated on a 4-qubit system (H_2, HeH^+, and LiH molecules) and 6-qubit system (H_3^- molecule). For the 8-qubit system, we examined the performance on the linear structure of H_4. Each optimizer minimized the variational parameters for the VHyP cost functions, with results shown in Figures 9 to 13.

- **Optimization Performance with H_2 systems at the bond length of $0.74\mathring{A}$ on a 4-qubit system with a single depth of HEA unitary transformation**

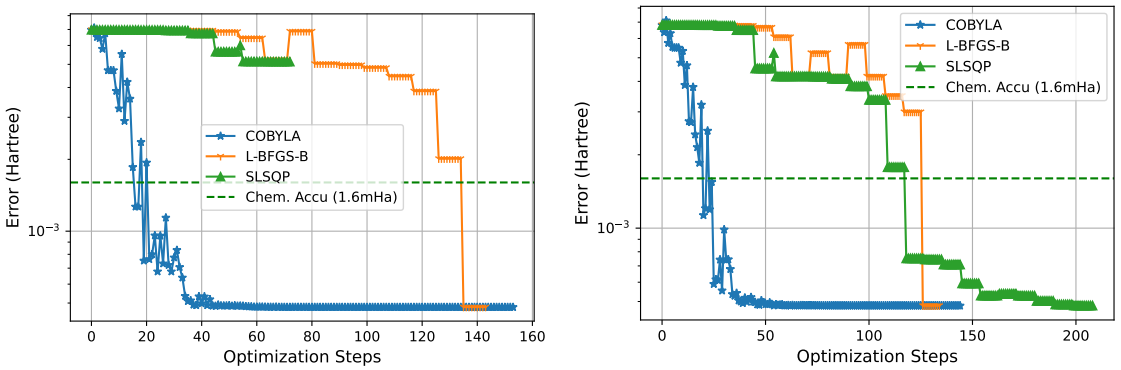


(a) Energy gain Cost Function Performance (b) Quantum QR Cost Function Performance

Figure 9: Optimization energy convergence for the both cost functions relative to FCI(Exact) on H_2 molecule;

The classical optimization performance tested with H_2 molecules shows a good accuracy with FCI result. L-BFGS-B converges more faster than COBYBY and SLSQP giving an error less than 10^{-12} Hartree for both cost functions. These results show that classical optimization algorithms work effortlessly with this approach.

- **Optimization Performance with HeH^+ systems at the bond length of $1.0\mathring{A}$ on a 4-qubit system with a single depth of HEA unitary transformation**



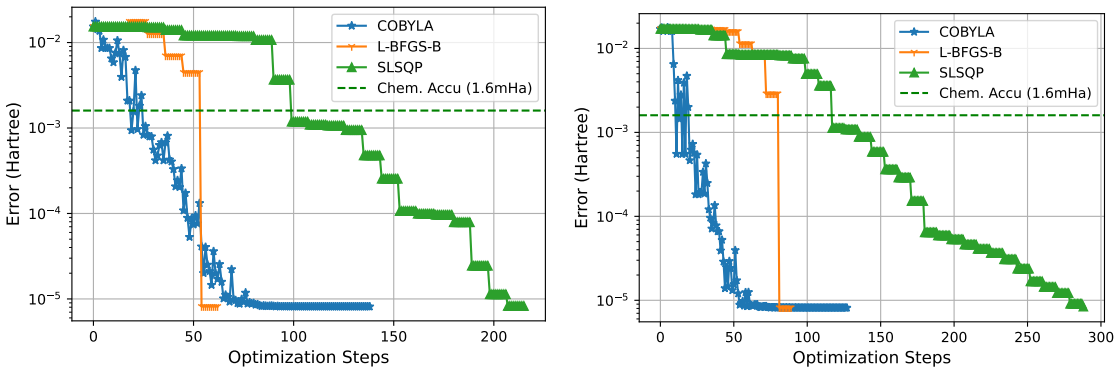
(a) Energy gain Cost Function Performance (b) Quantum QR Cost Function Performance

Figure 10: Optimization energy convergence for the both cost functions relative to FCI(Exact) on HeH^+ molecule;

The result of HeH^+ proves that this method preserves the symmetry by remaining in a

2-electron configuration throughout the simulation. This type of system is problematic for HEA-VQE because the simulation would end up in a 3-electron configuration. Here, the Quantum QR cost function shows fewer local minimum issues than the energy gain cost function with SLSQP optimizer.

- **Optimization Performance with LiH systems at the bond length of $1.5\mathring{A}$ on a 4-qubit system with a single depth of HEA unitary transformation**



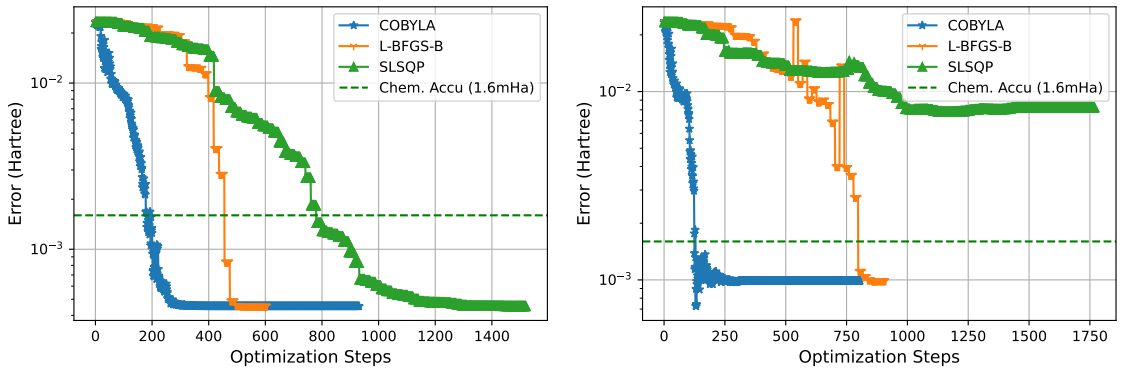
(a) Energy gain Cost Function Performance

(b) Quantum QR Cost Function Performance

Figure 11: Optimization energy convergence for the both cost functions relative to FCI(Exact) on LiH molecule;

The results of figure 9 to 11 are 2-electronic systems and simulated with a single depth of HEA of a 4-qubit system, which is composed of 6 single quantum rotation gates and 3 CNOT gates. This quantum computational resource requirement is minimal compared to other approaches, especially for LiH molecules. This also shows the efficiency of this approach with a high degree of accuracy.

- **Optimization Performance with H_3^- systems at the bond length of $1.0\mathring{A}$ on 6-qubit system with $depth = 2$ of HEA unitary transformation**

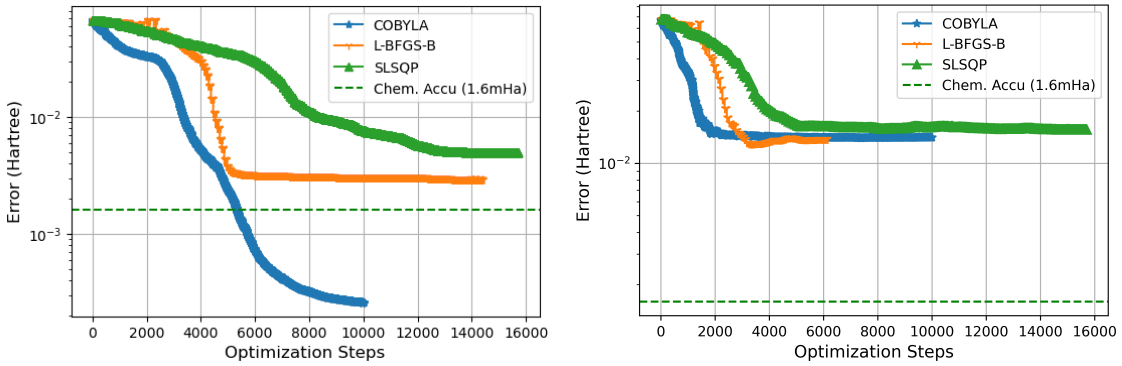


(a) Energy gain Cost Function Performance (b) Quantum QR Cost Function Performance

Figure 12: Optimization energy convergence for the both cost functions relative to FCI(Exact) on H_3^- molecule;

Here, The result shows a 6-qubit system with two depths of HEA to achieve chemical accuracy. This is also a test to prove that this approach indeed preserves symmetry and uses fewer resources.

- **Optimization Performance with H_4 systems at the bond length of $1.0\mathring{A}$ on a 8-qubits system with $depth = 9$ of HEA unitary transformation**



(a) Energy gain Cost Function Performance (b) Quantum QR Cost Function Performance

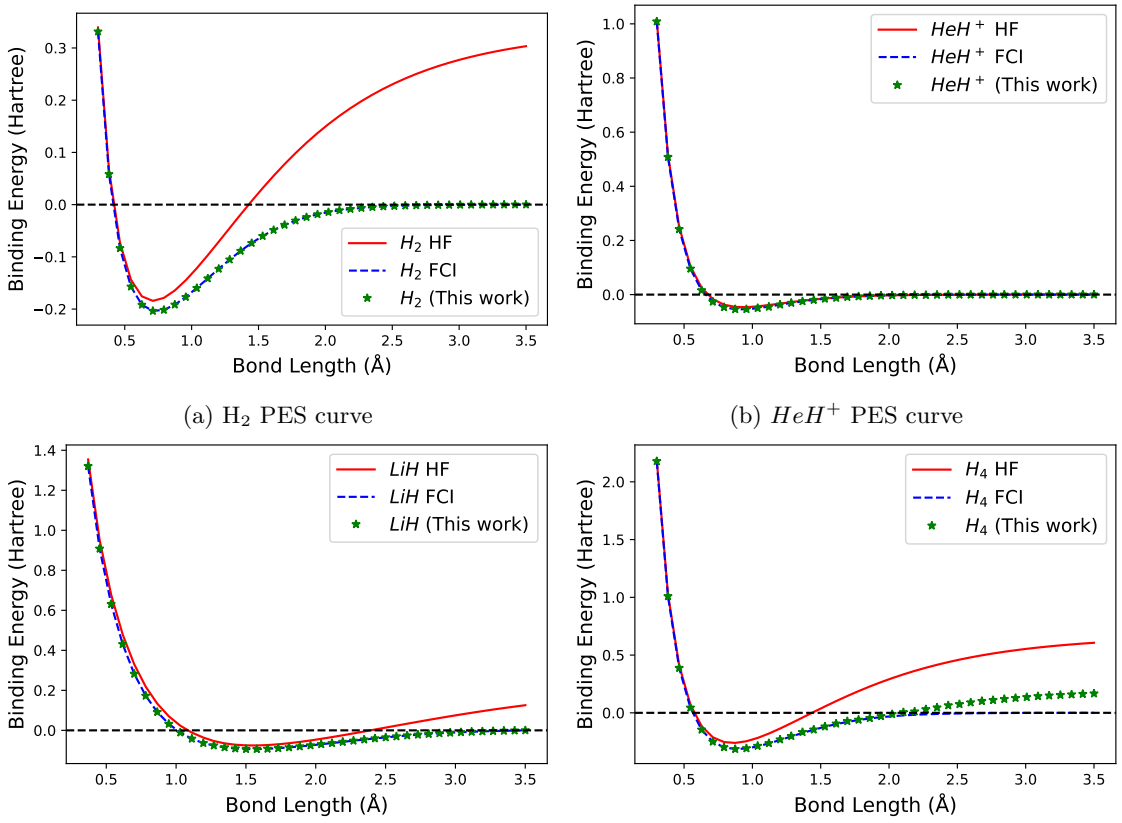
Figure 13: Optimization energy convergence for the both cost functions relative to FCI(Exact) on H_4 molecule.

These results revealed that the energy gain cost function and the Quantum QR cost function exhibit distinct convergence behaviors across molecules or the increasing number of qubit systems, with

COBYLA and L-BFGS-B optimizers generally outperforming SLSQP. Figures illustrate that, for most systems, L-BFGS-B converges faster than the rest. COBYLA achieves consistent convergence with a smaller number of iterations. Notably, the energy gain cost function yielded stable convergence even for deeper circuits in the VHyP method, whereas Quantum QR produced efficient results, particularly for systems with symmetry-preserved structures and for finding the energy spectrum of the system.

4.0.2 Potential Energy Surface (Dissociation Curves)

This approach was used to map the potential energy surfaces (PES) of H_2, HeH^+, LiH, and H_4. As shown in Figures 14, this method achieved chemical accuracy across the bond lengths tested except H_4 beyond the bond length of 2.0 Å, proving stable and reliable for ground-state energy calculations without misidentification of complementary states.



(a) H_2 PES curve

(b) HeH^+ PES curve

(c) LiH PES curve

(d) H_4 PES curve

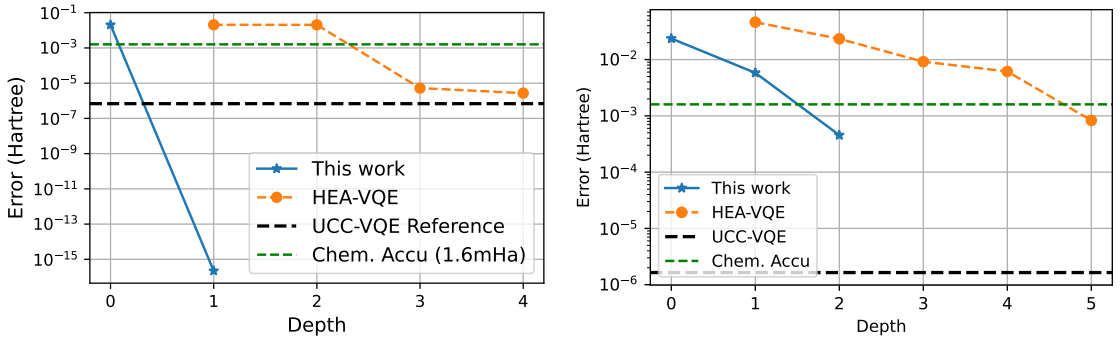
Figure 14: Binding energy as a function of the H-H inter-atomic distance in Å for H_2, HeH^+, LiH, and H_4 chains. We compare the state vector simulation of our algorithm with Hartree-Fock mean-field results and exact diagonalization (FCI).

These Results are obtained by minimizing at the state vector level of the energy gain cost function with a HEA ansatz of depth n and are compared to HF and FCI data. The binding energy is computed as $E_B = E_0 - E_D$, where E_D is the dissociation energy computed at the FCI level at 3.5 Å.

These PES results have proven that this concept is indeed efficient for solving the electronic structure problems of molecules using quantum computing. The behavior of the H_4 curve beyond 2.0Å is because toward the dissociation, the system requires more computational resources and this has been solved through the iteration algorithm.

4.0.3 Circuit layer and Gate Count

A key finding is that this algorithm achieved substantial savings in CNOT gate count and circuit layer compared to traditional HEA-VQE and UCC-VQE approaches. This is demonstrated in Figure 15. Using the Egain cost function, the VHyP circuit required fewer CNOT gates while reaching the desired chemical accuracy.



(a) HEA depth/layer comparison on H_2

(b) HEA depth/layer comparison on H_3^-

Figure 15: HEA depth/layer comparison of H_2 on 4-qubit and H_3^- on 6-qubit systems

We have shown that this method requires less computational resources to achieve a fidelity $F > 0.9999$. When comparing the accuracy of H_2, a 4-qubit system of a depth $(d = 1)$ of HEA, which contains 3 CNOT gates and 4 RY gates, we obtained a chemical accuracy of the order of 10^{-16} mHa. The HEA-VQE reaches the chemical accuracy of the order of 10^{-6} mHa at the depth $(d = 3)$, which consists of 9 CNOT gates and 16 RY gates as shown in figure 15a. The physically inspired UCC-VQE algorithms with 64 CNOT gates estimate the ground state energy at the chemical accuracy of the order of 10^{-6}. The optimization of H_3^- with HEA-VQE broke the symmetry in which the calculation converges to 3 electronic symmetries instead of 4 electronic symmetries, and this is as a result of the symmetry-breaking problem with HEA-VQE. These results have been tabulated in the table.

The detailed analysis in Table 1 compares the quantum resources required by this approach, HEA-VQE, and UCC-VQE, demonstrating that this method's efficient use of quantum resources results in lower computational overhead for achieving high fidelity and chemical accuracy.

Table 1: Comparison of different quantum circuit algorithm CNOT gate requirements to achieve chemical accuracy of 1.6 mHa for different molecules.

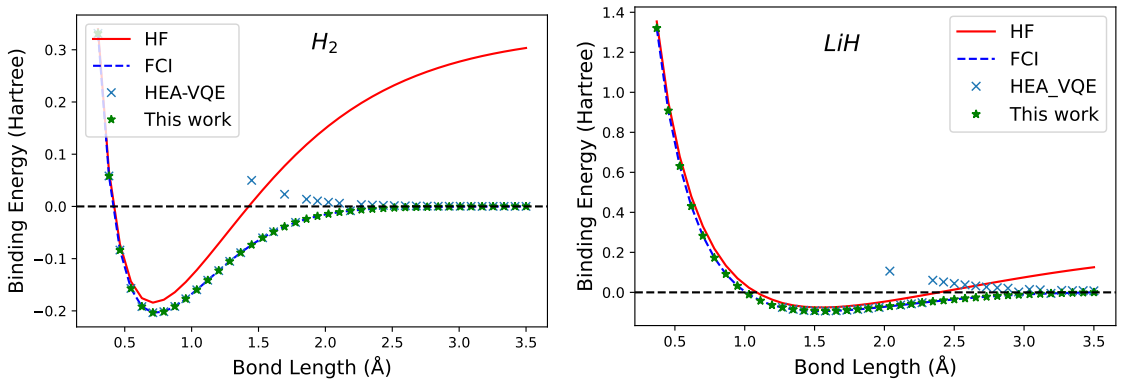
Molecule/Qubits	This work CNOT/Layer	HEA-VQE CNOT/Layer	UCC-VQE CNOT count
H_2 /4	3/1	9/3	64
HeH^+ /4	3/1	9/3	64
LiH /4	3/1	9/3	64
$H3^-$ /6	10/2	50/10	272
H_4 / 8	63/9	-	1312

The results show that the VHyP algorithm significantly reduces the quantum resource requirements, making it suitable for NISQ devices. By optimizing the cost function using the energy gain function, VHyP achieves high accuracy using a fewer number of gates, which helps suppress both the noise and computational overhead.

These results underline the importance of circuit efficiency in quantum algorithms. The reduced gate count of VHyP directly translates to increased performance on NISQ devices, where quantum decoherence and gate errors impose significant limitations. This approach is further capable of extending to larger molecules without requiring deep circuits, an improbable task with current quantum hardware.

4.0.4 Comparative Analysis of PES Using VHyP and HEA-VQE

A critical comparative analysis was conducted to evaluate the effectiveness of VHyP against HEA-VQE in obtaining the ground-state potential energy surface. Figures 16 reveal that VHyP accurately represents the ground state for H_2 and LiH, whereas HEA-VQE occasionally converges to excited states. This behavior aligns with previous findings [17] and validates this method as a superior method for ensuring accurate ground-state energy estimation.



(a) PES comparison using HEA on H_2 (b) PES comparison using HEA on LiH

Figure 16: PES comparison using HEA (4-qubit system) on H_2 and LiH

Additionally, this result highlights the crucial role that preservation of symmetry plays within quantum algorithms. By preserving the physical constraints of the problem, VHyP provides a more robust process of optimization and reduces the likelihood of convergence toward non-physical states. From this great advantage, VHyP may be an effective approach to more complex molecular systems where accurate ground-state estimates are critical to understanding chemical and physical phenomena.

4.0.5 Charge Gap Curve

The charge gap, $\Delta E_c = E_0(N_e + 1) + E_0(N_e - 1) - 2E_0(N_e)$, where $E_0(X)$ represents the system's ground state energy with X electrons, was computed for the H_2 and H_4 chains. At the mean-field level, ΔE_c represents the energy gap between the highest occupied and lowest unoccupied molecular orbitals, known as the HOMO-LUMO gap.

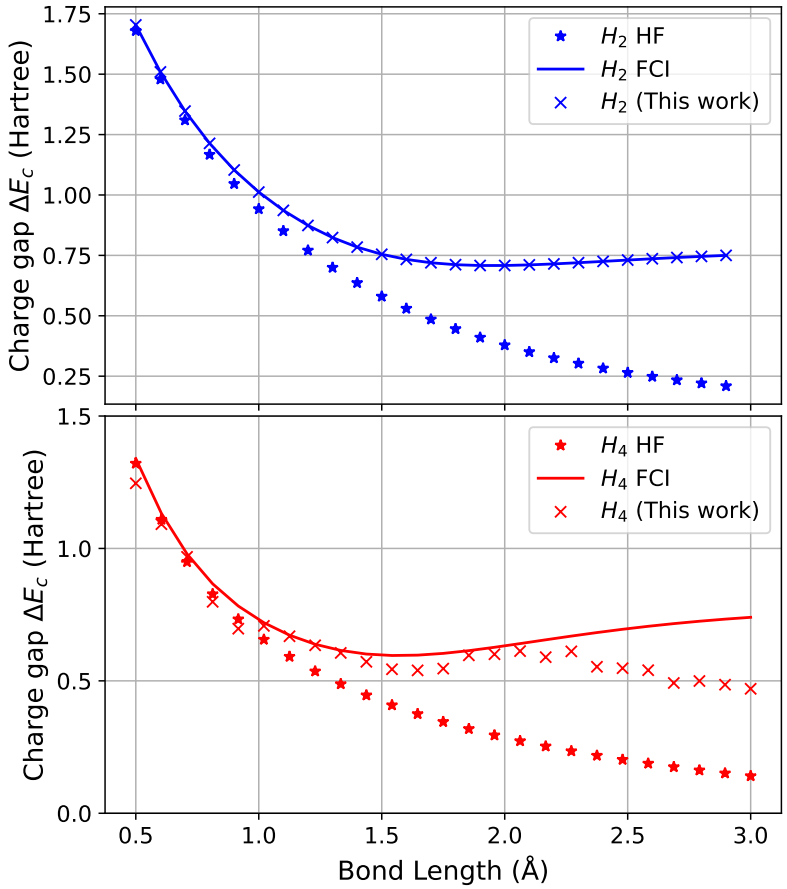


Figure 17: Charge gap as a function of the H-H inter-atomic distance in Å for H_2 and H_4 chains. We compare the state vector simulation of our algorithm with Hartree-Fock mean-field results and exact diagonalization (FCI).

The results in Figure 17 compare the VHyP approach to Hartree-Fock (HF) and Full Configuration Interaction (FCI). VHyP shows an excellent agreement for charge gap, except for H_4 molecules with H–H distances greater than 2 Å. The correct charge gap results show that utilizing Eq. (3.18) as a cost function enforces reduction to stay within the required symmetry-defined subspace. Using normal VQE with the HEA ansatz yields $\Delta E_c = 0$ for all H–H distances. In that instance, the optimization should result in the same global minimum, regardless of whether the electron number is fixed.

These results show the flexibility of VHyP in describing important quantum chemical properties, such as electron correlation and excitation energies. This degree of accuracy is particularly important for materials science applications, where the charge gap is often related to the electronic and optical properties of the materials.

4.0.6 Iterative Result

The dissociation curve of H_4 as shown in the figure 14d deviates from the FCI result beyond 2 Å. Investigating further shows that at these distances, $|\Phi_0\rangle$ is no longer a valid wavefunction. Indeed, ω^2, obtained by diagonalizing the optimized $\mathbf{H}^R(\boldsymbol{\theta}^*)$, saturates at 0.5 after only 3 layers as shown in the figure 18. This figure shows the accuracy, fidelity, and weight ω^2 as a function of the number of layers for H_4 chains with H–H bond distances of 1.0, 1.5, and 2.0 Å. For H–H bond distances of 1.0 and 1.5 Å, chemical accuracy of 1.6 mHartree is achieved at 7 and 9 layers, respectively. For an H–H bond distance of 2 Å, after 9 layers, the energy and fidelity do not improve with additional layers.

Thus, for initial states that are not a *good guess*, minimizing E_G favors large E_G which results in weight close to 0.5 by imposing $H_{00} < \widetilde{H}_{11}$. The energy associated to $|\Psi_0(\boldsymbol{\theta}^*)\rangle$ is nevertheless reduced compared to the initial energy of $|\Phi_0\rangle$ making $|\Psi_0(\boldsymbol{\theta}^*)\rangle$ a better *good guess* than $|\Phi_0\rangle$.

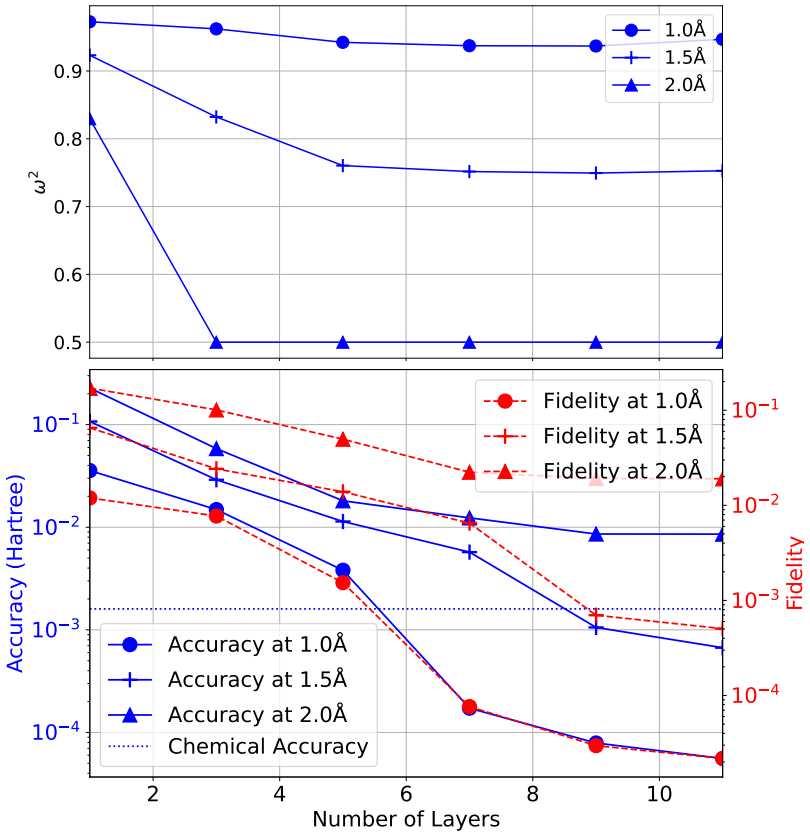


Figure 18: Weight ω^2 (top), accuracy and fidelity (bottom) as a function of the number of layers for H_4 chains with H-H distances of 1, 1.5 and 2 Å.

The results of the convergence of the iterative algorithm are shown in the figure 19 as a function of the iteration number for H_4 chains, where H atoms are separated by 1 and 2 Å. Different circuit depths, ranging from 1 to 9 layers, have been considered.

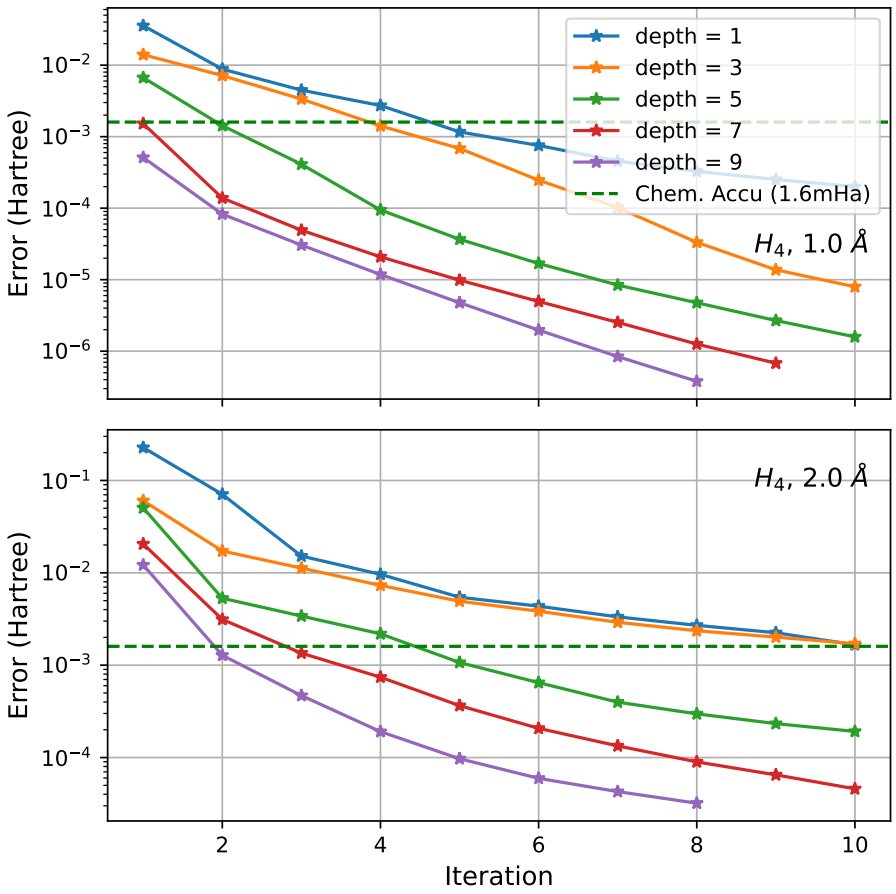


Figure 19: Error as a function of the iteration number for H_4 chains with H-H distances of 1 and 2 Å. Results are shown for different circuit depths corresponding to the number of layers in the HEA.

The observed error exhibits a rapid and systematic decrease, independent of the depth of the circuit. This observation indicates that chemical accuracy can be attained with a sufficient number of iterations, irrespective of the circuit depth. Furthermore, this methodology can also serve as error reduction method as a function of iteration number.

Results indicate that smaller circuits with more iterations are helpful to circumvent optimization problems. For example, for a 1-layer circuit, about 10 (5) iterations are needed to reach chemical accuracy with 8 variational parameters optimized per step, compared to a single optimization with 72 parameters in a 9-layer circuit at 2 Å (1 Å), respectively.

The iterative process used by VHyP demonstrates its ability to handle challenging situations where initial guesses are inadequate. By a systematic improvement of the guess state, VHyP ensures reliable energy predictions. The iterative approach will provide a practical means of quantum simulations, especially in such cases when optimal initial guesses are often unavailable. High accuracy

can be attained without the use of deep circuits, which reduces the hardware requirements and increases the method's applicability to different quantum chemistry problems.

5 Conclusion

This work presents and benchmarks the Variational Hyperspace Projection (VHyP) algorithm as a novel strategy for solving quantum chemistry problems on quantum computers. Avoiding the pitfalls of previous techniques, such as symmetry-breaking in HEA-VQE or high gate counts in UCC-VQE, VHyP provides a practical and feasible solution for noisy intermediate-scale quantum (NISQ) devices. The following conclusions are drawn from this study:

- **Optimization Performance**: VHyP uses optimization algorithms that yield robust convergence for a large class of molecular systems. The introduction of symmetry-preserved cost functions enhances reliability and accuracy for optimization.
- **Resource Efficiency**: The algorithm is significantly good at reducing the quantum resource requirements to achieve chemical accuracy using fewer CNOT gates and shallower circuits. This makes VHyP well-suited for implementation on current quantum hardware.
- **Physical Property Preservation**: Preserving physical constraints of the problem space, VHyP does not converge to non-physical states. Therefore, calculations of many-body systems are exact for properties such as the potential energy surface and charge gap.
- **Scalability and Adaptability**: The iterative approach followed by VHyP proves to be scalable and adaptive in complex systems. The ability to converge iteratively to refine initial guesses gives reliable results and gets rid of deep circuits.
- **Broad Applicability**: The success of VHyP in simulating molecular systems like H_2, HeH^+, LiH, H_3^-, and H_4 shows its very encouraging applicability in many different areas of quantum chemistry. Moreover, its ability to treat dissociation regions and preserve symmetry in structures proves its flexibility.

5.1 Future Directions

This study lays a platform for several lines of future research:

- **Implementation on Real Quantum Hardware**: Efforts are underway to benchmark VHyP on real quantum computing systems, which will further prove the functional applicability of the framework.
- **Extension to Larger Systems**: It is also important to extend the application of VHyP to larger molecular systems and to study its scalability with growing numbers of qubits.
- **Optimization Algorithm Refinement**: A deeper look into optimization algorithms, including quantum-inspired classical algorithms, could improve convergence rates and reduce computational costs.

5.2 Final Remarks

This work demonstrates the capabilities of the VHyP algorithm as a transformative tool in quantum chemistry for NISQ devices. Overcoming the strong constraints of conventional methodologies, VHyP not only increases the precision and efficiency of quantum simulations but also opens the way to the development of practical applications of quantum computing in scientific and industrial

contexts. With the improvement of quantum hardware, such methods as VHyP will become important in achieving the full potential of quantum computing for the solution of complicated problems arising in reality.

6 Acknowledgement

I am profoundly grateful to my supervisor, Prof. Matthieu Saubanère, for his precious teaching, advice, and support without ceasing during this project. His guidance has been important in my growing understanding and approach within this field.

I would like to express my sincere thanks to the Hubert Curien Program, financed by the Ministry of Europe and Foreign Affairs (CAMPUS FRANCE), for the mobility support that made this research possible. I would also like to thank the Laboratoire Ondes et Matière d'Aquitaine (LOMA)—CNRS for their kind hospitality and all the support provided during my research stay.

I would also like to take this opportunity to thank Abdus Salam ICTP for allowing me to access their high-performance computing facilities, without which my computational studies would not have been possible. I would also like to extend my gratitude to my collaborators, in particular, Dr. Bruno Senjean, for their support, stimulating discussions, and encouragement that have seriously enriched my knowledge in this field over the last few months.

I am especially very grateful to Prof. Catherine Meriaux, the acting director of ICTP-EAIFR, for support and motivation in that respect, without which this work could not have been completed. I am also much beholden to the ICTP-TWAS Fellowship Programme for full financial support during my master's studies at the University of Rwanda, which has enabled me to pursue my academic aspirations to fruition.

To my family, friends, and colleagues, your constant encouragement, guidance, and belief in me have been my pillars of strength throughout. Your presence has made this achievement all the more meaningful.

In the final analysis, I wish to express my most profound thanks to the Almighty God for blessing me with wisdom, knowledge, and understanding, besides the gift of life and the strength to complete this work.

7 AppendixA

7.1 Proof:

7.1.1 Energy gain in a two-level system

let consider the case where $H_{00} \leq H_{11}(\boldsymbol{\theta})$

$$\mathbf{H}^R(\boldsymbol{\theta}) = H_{00}1 + \begin{bmatrix} 0 & H_{10}(\boldsymbol{\theta}) \\ H_{01}(\boldsymbol{\theta}) & \Delta(\boldsymbol{\theta}) \end{bmatrix} \tag{7.1}$$

with $\Delta(\boldsymbol{\theta}) = H_{11}(\boldsymbol{\theta}) - H_{00} \geq 0$. The characteristic polynomial

$$\lambda^2 - \lambda\Delta(\boldsymbol{\theta}) - H_{01}(\boldsymbol{\theta})H_{10}(\boldsymbol{\theta}) = 0 \tag{7.2}$$

is equal to zero for the eigenvalues λ. It follows that the ground state of that 2 level system $E^{2L} = H_{00} + E_G$ where

$$E_G = \frac{\Delta(\boldsymbol{\theta})}{2}\left[1 - \sqrt{1 + \frac{4H_{01}(\boldsymbol{\theta})H_{10}(\boldsymbol{\theta})}{\Delta(\boldsymbol{\theta})^2}}\right] \tag{7.3}$$

where E_G (≤ 0) is the energy gained respect to H_{00} by the 2 level interaction, and is the function to be optimized such as minimizing the energy. for a week interacting system

$$E_G \simeq -\frac{H_{01}(\boldsymbol{\theta})H_{10}(\boldsymbol{\theta})}{\Delta(\boldsymbol{\theta})} \tag{7.4}$$

Now if $H_{11}(\boldsymbol{\theta}) \leq H_{00}$

$$\mathbf{H}^R(\boldsymbol{\theta}) = H_{11}(\boldsymbol{\theta})1 + \begin{bmatrix} \Delta(\boldsymbol{\theta} & H_{10}(\boldsymbol{\theta}) \\ H_{01}(\boldsymbol{\theta}) & 0 \end{bmatrix} \tag{7.5}$$

with $\Delta(\boldsymbol{\theta}) = |H_{11}(\boldsymbol{\theta}) - H_{00}| \geq 0$. The characteristic polynomial is equivalent than in the previous cases such that the energy gain respect to H_{11}, i.e. the energy lowering due to the interaction between the two level has the same form than Eq. (7.3). It follows that whatever $H_{11}(\boldsymbol{\theta})$ is higher or lower in energy than H_{00}, Eq. (7.3) can be use to evaluate the energy lowering due by the 2 level system considering $\Delta(\boldsymbol{\theta}) = |H_{11}(\boldsymbol{\theta}) - H_{00}|$.

7.1.2 Constrain contained cost function

This section consists of avoiding a constrained minimization by setting up the constrain $\mathbf{R}(\boldsymbol{\theta})|\Phi_0\rangle = |\Phi_0\rangle$ directly on the cost function. To that aim, we use that

$$\mathbf{R}(\boldsymbol{\theta})|\Phi_0\rangle = |\Phi_0\rangle \tag{7.6}$$

$$\langle\Phi_1|\mathbf{R}(\boldsymbol{\theta})|\Phi_0\rangle = \langle\Phi_1|\Phi_0\rangle \tag{7.7}$$

$$\langle\widetilde{\Phi}_1(\boldsymbol{\theta})|\Phi_0\rangle = 0. \tag{7.8}$$

It follows that we can use a Gramm-Shmidt orthonormalization process to constrain $|\widetilde{\Phi}_1(\boldsymbol{\theta})\rangle$ to be orthonormal to $|\Phi_0\rangle$ as

$$|\Phi_1^{\perp}(\boldsymbol{\theta})\rangle = \vartheta\left(z|\widetilde{\Phi}_1(\boldsymbol{\theta})\rangle - \sqrt{1-z^2}|\Phi_0\rangle\right), \tag{7.9}$$

with $\langle\Phi_0|\widetilde{\Phi}_1(\boldsymbol{\theta})\rangle = \vartheta|S|$, ($\vartheta = \pm 1$ and $|S|$ are the sign and amplitude of the overlap), $z = 1/\sqrt{1+S^2}$. The reduced Hamiltonian of the normalized two-level system from which the cost function is computed becomes

$$\mathbf{H}^{R\perp}(\boldsymbol{\theta}) = H_{00}1 + \begin{bmatrix} 0 & H_{10}^{\perp}(\boldsymbol{\theta}) \\ H_{01}^{\perp}(\boldsymbol{\theta}) & \Delta^{\perp}(\boldsymbol{\theta}), \end{bmatrix} \tag{7.10}$$

with $H_{10}^{\perp}(\boldsymbol{\theta}) = \langle\Phi_0|\hat{\mathbf{H}}|\widetilde{\Phi}_1^{\perp}(\boldsymbol{\theta})\rangle$ and $H_{11}^{\perp}(\boldsymbol{\theta}) = \langle\widetilde{\Phi}_1^{\perp}(\boldsymbol{\theta})|\hat{\mathbf{H}}|\widetilde{\Phi}_1^{\perp}(\boldsymbol{\theta})\rangle$, $\Delta^{\perp}(\boldsymbol{\theta}) = |H_{11}^{\perp}(\boldsymbol{\theta}) - H_{00}|$.

7.1.3 Triangularization of the Hamiltonian

In this section we show that maximizing $H_{01}(\boldsymbol{\theta})H_{10}(\boldsymbol{\theta})$ with the condition that $\langle\Phi_0|\mathbf{R}(\boldsymbol{\theta})\Phi_0\rangle = 1$ consist in triangularizing the Hamiltonian.
The condition $\langle\Phi_0|\mathbf{R}(\boldsymbol{\theta})\Phi_0\rangle = 1$ impose that Let consider now the 00 element of the square of the Hamiltonian in the Householder and variational representations, it follows

$$[\widetilde{H}\widetilde{H}]_{00} = H_{00}^2 + \widetilde{H}_{01}\widetilde{H}_{10} \tag{7.11}$$

$$[H(\boldsymbol{\theta})H(\boldsymbol{\theta})]_{00} = H_{00}^2 + H_{01}(\boldsymbol{\theta})H_{10}(\boldsymbol{\theta}) + \sum_{j>1} H_{0j}(\boldsymbol{\theta})H_{j0}(\boldsymbol{\theta}). \tag{7.12}$$

However, the condition $\langle\Phi_0|\mathbf{R}|(\boldsymbol{\theta})\Phi_0\rangle = 1$ impose that $R_{0j}(\boldsymbol{\theta}) = R_{j0}(\boldsymbol{\theta}) = \delta_{j0}$ and consequently that $[\widetilde{H}\widetilde{H}]_{00} = [H(\boldsymbol{\theta})H(\boldsymbol{\theta})]_{00}$. Consequently,

$$\widetilde{H}_{01}\widetilde{H}_{10} > H_{01}(\boldsymbol{\theta})H_{10}(\boldsymbol{\theta}) \tag{7.13}$$

except if for all $j > 1$, $H_{j0}(\boldsymbol{\theta}) = 0$. In that latter case, $\widetilde{H}_{01}\widetilde{H}_{10} = H_{01}(\boldsymbol{\theta})H_{10}(\boldsymbol{\theta})$. Consequently, optimizing $\boldsymbol{\theta}$ to maximize $H_{01}(\boldsymbol{\theta})H_{10}(\boldsymbol{\theta})$ cancel out all non diagonal term of the 0 column of $\mathbf{H}(\boldsymbol{\theta})$.

7.2 Code spinets:

7.2.1 Libraries and Dependencies

Importing necessary libraries to construct and run the simulation. These libraries allow for the use of tools that help in building quantum circuits, manipulating molecular Hamiltonians, and running classical optimization algorithms.

```
from typing import Optional, List
import numpy as np
import warnings
from qat.core import Observable, Circuit
from qat.lang.AQASM import *
from qat.lang.AQASM.gates import Gate
from qat.core import Term, Observable
from qat.fermion.hamiltonians import FermionHamiltonian
from qat.qpus import get_default_qpu
from qat.plugins import ScipyMinimizePlugin
from qat.fermion.hamiltonians import make_hubbard_model
from qat.fermion.chemistry.pyscf_tools import perform_pyscf_computation
```

```
from qat.fermion import SpinHamiltonian
from qat.qpus import PyLinalg, get_default_qpu
from qat.fermion.chemistry import MolecularHamiltonian, MoleculeInfo
from qat.fermion.transforms import *
import matplotlib.pyplot as plt
from scipy.optimize import minimize
```

These libraries allow us to build quantum circuits, to calculate molecular Hamiltonians, and to optimize classically in the ground state energy search.

7.2.2 Geometry and Molecular Hamiltonian Setup

We start by defining the geometry of the molecule under study. Here, we take the hydrogen molecule H_2 as an example. We also define the basis set and other parameters such as spin and charge. The molecular integrals for this setting are calculated using the PySCF package.

```
geometry = [
    ("H", (0.0, 0.0, 0.0)),
    ("H", (0.0, 0.0, 1.0))
]
basis = "sto-3g"
spin = 0
charge = 0
( rdm1, orbital_energies, nuclear_repulsion, n_electrons, one_body_integrals,
two_body_integrals, info,) = perform_pyscf_computation(geometry=geometry,
basis=basis, spin=spin, charge=charge, run_fci=True)

print(f"HF energy: {info['HF']}\nMP2 energy: {info['MP2']}\n
FCI energy: {info['FCI']}\n")
print(f"Number of qubits = {rdm1.shape[0] * 2}")
```

The molecular Hamiltonian $\mathbf{H}$ for a system of electrons in a set of orbitals is given by:

$$\mathbf{H} = \sum_{ij} h_{ij} a_i^\dagger a_j + \frac{1}{2} \sum_{ijkl} g_{ijkl} a_i^\dagger a_j^\dagger a_k a_l, \tag{7.14}$$

where h_{ij} represents the one-electron integrals, g_{ijkl} represents the two-electron integrals, and $a_i^\dagger$, a_i are the fermionic creation and annihilation operators, respectively.

The function 'perform-pyscf-computation' calculates these integrals and returns the Hartree-Fock (HF), MP2, and Full Configuration Interaction (FCI) energies. The integrals are necessary for constructing the molecular Hamiltonian.

7.2.3 Hamiltonian Transformation to Spin Basis

The molecular Hamiltonian is constructed from the one-body and two-body integrals, then transformed to the Jordan-Wigner spin basis for quantum simulation.

```
# Define the molecular Hamiltonian
mol_h = MolecularHamiltonian(one_body_integrals, two_body_integrals,
nuclear_repulsion)

# Compute the ElectronicStructureHamiltonian
H_ele = mol_h.get_electronic_hamiltonian()
H_spin = transform_to_jw_basis(H_ele)
H_matrix = H_spin.get_matrix()
H_mat_r = np.real(H_matrix)
eigenvalues, eigenvectors = np.linalg.eig(H_matrix)
min_index = np.argmin(eigenvalues)
Fci_state = eigenvectors[:, min_index]
```

In the spin basis, the Hamiltonian $\mathbf{H}$ can be represented as:

$$\mathbf{H} = \sum_{\alpha\beta} h_{\alpha\beta}\sigma_\alpha^\dagger\sigma_\beta + \frac{1}{2}\sum_{\alpha\beta\gamma\delta} g_{\alpha\beta\gamma\delta}\sigma_\alpha^\dagger\sigma_\beta^\dagger\sigma_\gamma\sigma_\delta, \tag{7.15}$$

where σ_α are the Pauli operators in the Jordan-Wigner (JW) transformed basis. H_{matrix} is the Hamiltonian in this basis and the lowest eigenvalue is the ground state energy.

7.2.4 Gate Definitions and Parameter Calculation

We use parameterized gates like R_y, R_x, U_2, and U_3 to construct the variational quantum circuit. To calculate the number of variational parameters used, we define the following function 'calculate-parameters'.

```
def calculate_parameters(n_qubits, n_layers, layer_gate, init_gate=None):
    n_params_init_gate = 0
    n_params_layer_gate = 0

    if init_gate == RY or init_gate == RX:
        n_params_init_gate = n_qubits
    elif init_gate == U2:
        n_params_init_gate = n_qubits * 2
    elif init_gate == U3:
        n_params_init_gate = n_qubits * 3

    if layer_gate == RY or layer_gate == RX:
        n_params_layer_gate = n_qubits * n_layers
    elif layer_gate == U2:
        n_params_layer_gate = n_qubits * n_layers * 2
    elif layer_gate == U3:
        n_params_layer_gate = n_qubits * n_layers * 3

    return n_params_init_gate + n_params_layer_gate, n_params_init_gate,
    n_params_layer_gate
```

The total number of variational parameters, n_{params}, is calculated as:

$$n_{\text{params}} = n_{\text{init\_gate}} + n_{\text{layer\_gate}}, \tag{7.16}$$

where $n_{\text{init\_gate}}$ and $n_{\text{layer\_gate}}$ are the number of parameters needed for the initialization and layer gates, respectively.

7.2.5 Complementary State Generation

We initialize qubits in a complementary state by using the 'generate-excited-state' function, which allows the excitation mode to be specified as "sparse," "full," or "random."

```
def generate_excited_state(n_qubits: int, mode: str = "sparse") -> List[int]:
    if mode == "sparse":
        excited_state = [1 if i % 2 == 0 else 0 for i in range(n_qubits)]
    elif mode == "full":
        excited_state = [1] * n_qubits
    elif mode == "random":
        excited_state = np.random.choice([0, 1], size=n_qubits).tolist()
    return excited_state
```

The excited state vector can be represented as:

$$|\text{Exs}\rangle = \bigotimes_{i=0}^{n_{\text{qubits}}-1} |s_i\rangle, \tag{7.17}$$

where $s_i \in \{0, 1\}$, determined by the chosen mode ("sparse," "full," or "random").

7.2.6 Hartree-Fock Circuit

The Hartree-Fock state is an essential starting point for quantum chemistry simulations. We can construct the HF state via the build-HF-circuit function.

```
def build_HF_circuit(n_qubits, n_electrons) -> Circuit:
    prog = Program()
    qbits = prog.qalloc(n_qubits)
    for i in range(n_electrons):
        prog.apply(X, qbits[i])
    return prog.to_circ()
```

The Hartree-Fock state $|\text{HF}\rangle$ is represented as:

$$|\text{HF}\rangle = \bigotimes_{i=0}^{n_{\text{electrons}}-1} |1\rangle_i \bigotimes_{i=n_{\text{electrons}}}^{n_{\text{qubits}}-1} |0\rangle_i. \tag{7.18}$$

This state represents the ground state configuration of the electrons in the system.

7.2.7 Quantum Circuit Construction

To build the quantum circuit, we start with the qubits in an excited state and then apply a parameterized unitary transformation. The following 'build-circuit-1' function constructs the circuit:

```
def build_circuit_1(params, n_qubits: int, init_gate: Gate, excited_state:
List[int], n_layers: int, layer_gate: Gate) -> Circuit:
    prog = Program()
    qbits = prog.qalloc(n_qubits)
    param_idx = 0
    if init_gate == RY or init_gate == RX or init_gate == U2 or init_gate == U3:
        for i, state in enumerate(excited_state):
            if state == 1:
                if init_gate == RY or init_gate == RX:
                    prog.apply(init_gate(params[param_idx]), qbits[i])
                    param_idx += 1
                elif init_gate == U2:
                    prog.apply(U2(params[param_idx], params[param_idx + 1]),
                    qbits[i])
                    param_idx += 2
                elif init_gate == U3:
                    prog.apply(U3(params[param_idx], params[param_idx + 1],
                    params[param_idx + 2]), qbits[i])
                    param_idx += 3
    else:
        for i, state in enumerate(excited_state):
            if state == 1:
                prog.apply(init_gate, qbits[i])
    u_qr = create_qroutine(params[param_idx:], n_qubits, n_layers, layer_gate)
    prog.apply(u_qr, qbits)
    return prog.to_circ()
```

The variational complementary wavefunction $|\widetilde{\Phi}_1\rangle$ is represented as:

$$|\widetilde{\Phi}_1\rangle = U(\boldsymbol{\theta})|\Phi_1\rangle, \tag{7.19}$$

where $U(\boldsymbol{\theta})$ is the unitary transformation parameterized by $\boldsymbol{\theta}$, applied to the state $|\Phi_1\rangle$.

7.2.8 Expectation Value Computation

The expectation values of the Hamiltonian are computed for a given trial wavefunction using the 'compute-expectation-values-1' function.

```
def compute_expectation_values_1(params, n_qubits, init_gate, excited_state,
n_layers, layer_gate):
    circuit_1 = build_circuit_1(params, n_qubits, init_gate, excited_state,
    n_layers, layer_gate)
```

```
    job_1 = circuit_1.to_job(job_type='statevector')
    qpu_1 = get_default_qpu()
    result_1 = qpu_1.submit(job_1)
    if result_1 is None or result_1.statevector is None:
        raise ValueError("Failed to retrieve the initial state vector from
-------the QPU result.")

    Ex = result_1.statevector/np.linalg.norm(result_1.statevector)
    b = np.dot(np.conj(HF_state), Ex)
    a = 1 / np.sqrt(b**2 + 1)
    orth_Ex = a * Ex - (b / np.sqrt(b**2 + 1)) * HF_state
    Ex1 = orth_Ex / np.linalg.norm(orth_Ex)

    E_00 = np.dot(HF_state.conj().T, np.dot(H_matrix, HF_state))
    E_01 = np.dot(HF_state.conj().T, np.dot(H_matrix, Ex1))
    E_10 = np.dot(Ex1.conj().T, np.dot(H_matrix, HF_state))
    E_11 = np.dot(Ex1.conj().T, np.dot(H_matrix, Ex1))

    return E_00, E_01, E_10, E_11, Ex1
```

The expectation values of the Hamiltonian in the basis of the Hartree-Fock state $|\text{HF}\rangle$ and the orthonormalized excited state $|\text{Ex1}\rangle$ are given by:

$$E_{00} = \langle \text{HF}|\mathbf{H}|\text{HF}\rangle, \tag{7.20}$$
$$E_{01} = \langle \text{HF}|\mathbf{H}|\text{Ex1}\rangle, \tag{7.21}$$
$$E_{10} = \langle \text{Ex1}|\mathbf{H}|\text{HF}\rangle, \tag{7.22}$$
$$E_{11} = \langle \text{Ex1}|\mathbf{H}|\text{Ex1}\rangle. \tag{7.23}$$

7.2.9 Cost Function and Optimization

The cost function minimizes the energy and maximizes the overlap with the true ground state. We use optimization methods such as COBYLA, SLSQP, or L-BFGS-B to minimize the energy.

```
def objective_function_1(params, n_qubits, init_gate, excited_state,
n_layers, layer_gate):
    E_00, E_01, E_10, E_11, phi_tidal_1 = compute_expectation_values_1(params,
    n_qubits, init_gate, excited_state, n_layers, layer_gate)
    E_matrix = np.array([[E_00, E_01], [E_10, E_11]])
    eigenvalues, eigenvectors = np.linalg.eig(E_matrix)
    min_eigenvalue_index = np.argmin(eigenvalues)
    lowest_eigenvector = eigenvectors[:, min_eigenvalue_index]
    trial_state = lowest_eigenvector[0] * HF_state + lowest_eigenvector[1] *
    phi_tidal_1
    psi_0 = trial_state / np.linalg.norm(trial_state)
    fidelity = np.linalg.norm(np.dot(Fci_state, psi_0))
```

```
    w = compute_weight(HF_state, psi_0)
    delta = np.abs(E_11 - E_00)
    cost_1 = -(delta/2)*(1 - np.sqrt(1 + (4*E_01*E_10)/delta**2 )) #Energy gain
    cost_2 = E_01 * E_10
    Energy_gs = (np.dot(np.conj(psi_0), np.dot(H_matrix, psi_0))).real
    return -cost_1, -cost_2, Energy_gs, fidelity
```

The objective function is constructed as a weighted sum of two cost functions:

$$\text{Cost}_1 = \frac{\Delta(\boldsymbol{\theta})}{2}\left[1 - \sqrt{1 + \frac{4H_{01}(\boldsymbol{\theta})H_{10}(\boldsymbol{\theta})}{\Delta(\boldsymbol{\theta})^2}}\right], \tag{7.24}$$

$$\text{Cost}_2 = E_{01} \times E_{10}, \tag{7.25}$$

where $w = \langle \text{HF} | \psi_{\text{trial}} \rangle$ is the overlap of the trial state with the Hartree-Fock state.

7.2.10 Optimization Process

Optimization is performed over a number of iterations using classical optimizers. Results are saved at every iteration, and the process is stopped once convergence is achieved.

```
iterations = 20
for iteration in range(1, iterations + 1):
    print(f"Starting iteration {iteration}")
    initial_params = np.random.rand(n_params) * np.pi
    if iteration == 1:
        result = minimize(lambda params: objective_function_1(params,
        n_qubits, init_gate, excited_state, n_layers, layer_gate)[0],
                          initial_params, bounds=bounds, method=method)
        optimized_params_1 = result.x
    else:
        result = minimize(lambda params: objective_function_iteration(params,
        n_qubits, init_gate, excited_state, n_layers, layer_gate, psi_current,
        Ex1_current, w_current)[0],
                          initial_params, bounds=bounds, method=method)
        optimized_params = result.x
```

This iterative process optimizes the parameters of the quantum circuit until the minimum energy is achieved.

References

[1] Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J Love, Alán Aspuru-Guzik, and Jeremy L O'Brien. A variational eigenvalue solver on a photonic quantum processor. *Nature communications*, 5:4213, 2014.

[2] Richard P Feynman. Simulating physics with computers. *International journal of theoretical physics*, 21(6-7):467–488, 1982.

[3] Peter W Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In *Proceedings 35th annual symposium on foundations of computer science*, pages 124–134. IEEE, 1994.

[4] Lov K Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 212–219, 1996.

[5] Wojciech H Zurek. Decoherence and the transition from quantum to classical. *Physics Today*, 44(10):36–44, 1991.

[6] Peter W Shor. Scheme for reducing decoherence in quantum computer memory. *Physical review A*, 52(4):R2493, 1995.

[7] John M. Martinis. Qubit metrology for building a fault-tolerant quantum computer. *Arxiv*, 2015.

[8] David J. Reilly. Challenges in scaling-up the control interface of a quantum computer. *2019 IEEE International Electron Devices Meeting (IEDM)*, pages 31.7.1–31.7.6, 2019.

[9] John Preskill. Quantum Computing in the NISQ era and beyond. *Quantum*, 2:79, August 2018.

[10] Attila Szabo and Neil S Ostlund. *Modern quantum chemistry: introduction to advanced electronic structure theory*. Courier Corporation, 2012.

[11] Rodney J Bartlett and Monika Musiał. Coupled-cluster theory in quantum chemistry. *Reviews of Modern Physics*, 79(1):291–352, 2007.

[12] P.J. Knowles and N.C. Handy. A new determinant-based full configuration interaction method. *Chemical Physics Letters*, 111(4):315–321, 1984.

[13] A.Yu.Kitaev. Quantum measurements and the abelian stabilizer problem. *Arxiv*, 1995.

[14] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010.

[15] Luca Pezze and Augusto Smerzi. *Quantum Phase Estimation Algorithm with Gaussian Spin States*. APS, 2021.

[16] Nikolaj Moll, Panagiotis Barkoutsos, Lev S Bishop, Jerry M Chow, Andrew Cross, Daniel J Egger, Stefan Filipp, Andreas Fuhrer, Jay M Gambetta, Marc Ganzhorn, et al. Quantum optimization using variational algorithms on near-term quantum devices. *Quantum Science and Technology*, 3(3):030503, 2018.

[17] Abhinav Kandala, Antonio Mezzacapo, Kristan Temme, Maika Takita, Markus Brink, Jerry M Chow, and Jay M Gambetta. Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature*, 549(7671):242–246, 2017.

[18] Peter O'Malley et al. Scalable quantum simulation of molecular energies. *Physical Review X*, 6(3):031007, 2016.

[19] Jonathan Romero, Ryan Babbush, Jarrod R McClean, Cornelius Hempel, Peter J Love, and Alán Aspuru-Guzik. Strategies for quantum computing molecular energies using the unitary coupled cluster ansatz. *Quantum Science and Technology*, 4(1):014008, 2018.

[20] Panagiotis Barkoutsos, Jeremy F Gonthier, Pauline J Ollitrault, and et al. Quantum algorithms for electronic structure calculations: Particle-hole hamiltonian and optimized wavefunction expansions. *Physical Review A*, 98(2):022322, 2018.

[21] Harper R Grimsley, Daniel Claudino, Sophia E Economou, Edwin Barnes, and Nicholas J Mayhall. An adaptive variational algorithm for exact molecular simulations on a quantum computer. *Nature Communications*, 10(1):3007, 2019.

[22] Jarrod R McClean, Jonathan Romero, Ryan Babbush, and Alán Aspuru-Guzik. The theory of variational hybrid quantum-classical algorithms. *New Journal of Physics*, 18(2):023023, 2016.

[23] Jarrod R McClean, Sergio Boixo, Vadim N Smelyanskiy, Ryan Babbush, and Hartmut Neven. Barren plateaus in quantum neural network training landscapes. *Nature Communications*, 9:4812, 2018.

[24] M Cerezo, Akira Sone, Tyler Volkoff, Lukasz Cincio, and Patrick J Coles. Cost function dependent barren plateaus in shallow parametrized quantum circuits. *Nature Communications*, 12(1):1791, 2021.

[25] John Preskill. Quantum Computing in the NISQ era and beyond. *Quantum*, 2:79, 2018.

[26] Kazuhiro Seki, Tomonori Shirakawa, and Seiji Yunoki. Symmetry-adapted variational quantum eigensolver. *Physical Review A*, 101(5):052340, 2020.

[27] Wataru Mizukami, Kosuke Mitarai, Yuya O. Nakagawa, Takahiro Yamamoto, Tennin Yan, and Yu-ya Ohnishi. Orbital optimized unitary coupled cluster theory for quantum computer. *Phys. Rev. Res.*, 2:033421, Sep 2020.

[28] Saad Yalouz Martin Beseda, Silvie Illesova and Bruno Senjean. State-averaged orbital-optimized vqe: A quantum algorithm for the democratic description of ground and excited electronic states. *Journal of Open Source Software*, 2024.

[29] Abhinav Kandala, Kristan Temme, Antonio D Córcoles, Antonio Mezzacapo, Jerry M Chow, and Jay M Gambetta. Error mitigation extends the computational reach of a noisy quantum processor. *Nature*, 567(7749):491–495, 2019.

[30] Kristan Temme, Sergey Bravyi, and Jay M. Gambetta. Error mitigation for short-depth quantum circuits. *Phys. Rev. Lett.*, 119:180509, Nov 2017.

[31] Pierre-Luc Dallaire-Demers, Jonathan Romero, Libor Veis, Sukin Sim, and Alán Aspuru-Guzik. Low-depth circuit ansatz for preparing correlated fermionic states on a quantum computer. *Quantum Science and Technology*, 4(4):045005, aug 2019.

[32] Cristian L. Cortes and Stephen K. Gray. Quantum Krylov subspace algorithms for ground- and excited-state energy estimation. *Phys. Rev. A*, 105(2):022417, February 2022.

[33] Nobuyuki Yoshioka, Hideaki Hakoshima, Yuichiro Matsuzaki, Yuuki Tokunaga, Yasunari Suzuki, and Suguru Endo. Generalized quantum subspace expansion. *Phys. Rev. Lett.*, 129:020502, Jul 2022.

[34] Alston S Householder. Unitary triangularization of a nonsymmetric matrix. *Journal of the ACM (JACM)*, 5(4):339–342, 1958.

[35] Nicholas J Higham. Matrix nearness problems and applications. *Applications of matrix theory*, 22:1–27, 1989.

[36] Alan Aspuru-Guzik, Anthony D Dutoi, Peter J Love, and Martin Head-Gordon. Simulated quantum computation of molecular energies. *Science*, 309(5741):1704–1707, 2005.

[37] James D Whitfield, Jacob Biamonte, and Alán Aspuru-Guzik. Simulation of electronic structure hamiltonians using quantum computers. *Molecular Physics*, 109(5):735–750, 2011.

[38] Dmitry A Fedorov, Bo Peng, Niranjan Govind, and Yuri Alexeev. Vqe method: A short survey and recent developments. *Materials Theory*, 2022. Accessed from user's provided document.

[39] Jarrod R. McClean, Mollie E. Kimchi-Schwartz, Jonathan Carter, and Wibe A. de Jong. Hybrid quantum-classical hierarchy for mitigation of decoherence and determination of excited states. *Phys. Rev. A*, 95:042308, Apr 2017.

[40] Mario Motta, Chenyang Sun, Adrian TKL Tan, Michael J O'Rourke, Erhai Ye, Austin J Minnich, Fernando GD Brandão, and Garnet Kin-Lic Chan. Determining eigenstates and thermal states on a quantum computer using quantum imaginary time evolution. *Nature Physics*, 16(2):205–210, 2020.

[41] Nicholas H Stair, Renke Huang, and Francesco A Evangelista. A multireference quantum krylov algorithm for strongly correlated electrons. *The Journal of Chemical Physics*, 16:2236–2245, 2020.

[42] Francisco Orts, Gloria Ortega, and Ester M. Garzón. Studying the Cost of n-qubit Toffoli Gates. In Derek Groen, Clélia de Mulatier, Maciej Paszynski, Valeria V. Krzhizhanovskaya, Jack J. Dongarra, and Peter M. A. Sloot, editors, *Computational Science – ICCS 2022*, pages 122–128, Cham, 2022. Springer International Publishing.

[43] Baptiste Claudon, Julien Zylberman, César Feniou, Fabrice Debbasch, Alberto Peruzzo, and Jean-Philip Piquemal. Polylogarithmic-depth controlled-NOT gates without ancilla qubits. *Nat Commun*, 15(1):5886, July 2024.

[44] Mario Motta, William Kirby, Ieva Liepuoniute, Kevin J. Sung, Jeffrey Cohn, Antonio Mezzacapo, Katherine Klymko, Nam Nguyen, Nobuyuki Yoshioka, and Julia E. Rice. Subspace methods for electronic structure simulations on quantum computers. *Electron. Struct.*, 6(1):013001, March 2024.

[45] M. J. D. Powell. A direct search optimization method that models the objective and constraint functions by linear interpolation. *Advances in Optimization and Numerical Analysis*, pages 51–67, 1994.

[46] Dieter Kraft. A software package for sequential quadratic programming. Technical Report DFVLR-FB 88-28, DFVLR - German Aerospace Center, Institute for Flight Mechanics, 1988.

[47] Ciyou Zhu, Richard H. Byrd, Peihuang Lu, and Jorge Nocedal. Algorithm 778: L-bfgs-b: Fortran subroutines for large-scale bound-constrained optimization. *ACM Transactions on Mathematical Software*, 23(4):550–560, 1997.

[48] Atos. *myQLM: Atos Quantum Learning Machine Development Toolkit*, 2016. Available online at https://myqlm.github.io/01_getting_started.html.

[49] Pascual Jordan and Eugene Wigner. Über das paulische Äquivalenzverbot. *Zeitschrift für Physik*, 47(9-10):631–651, 1928.