universität wien

MASTERARBEIT / MASTER'S THESIS

Titel der Masterarbeit / Title of the Master's Thesis

„A First Quantization Approach to the Variational Quantum Eigensolver“

verfasst von / submitted by

Leander Thiessen, BSc

angestrebter akademischer Grad / in partial fulfilment of the requirements for the degree of

Master of Science (MSc)

Wien, 2023 / Vienna, 2023

Studienkennzahl lt. Studienblatt / degree programme code as it appears on the student record sheet:	UA 066 876
Studienrichtung lt. Studienblatt / degree programme as it appears on the student record sheet:	Masterstudium Physics UG2002
Betreut von / Supervisor:	Ass.-Prof. Mag. Dr. Borivoje Dakić

Acknowledgements

First and foremost, I would like to thank Borivoje Dakić for his support and supervision of this thesis. In this project we pursued many different paths and changed directions several times along the way. This would not have been possible without your guidance and constant encouragement, especially at times when I felt like I could not make any progress. Thank you for the interesting discussions through which I learned a lot about how to explore and test scientific ideas.

I also want to thank Matteo Rossi and Guillermo García-Pérez from Algorithmiq for engaging in discussions and providing feedback that was essential in the development of this thesis.

A big thank you goes to Emma Neto for her constant support and encouragement throughout every part of the thesis. Thanks for keeping me optimistic, for providing necessary distractions, and for reading and correcting this thesis.

Abstract

Performing accurate quantum chemistry simulations is one of the main potential applications of near-term quantum computers. One of the most prominent algorithms for this purpose is the Variational Quantum Eigensolver (VQE), which allows us to predict ground states of molecular Hamiltonians. Although VQE does, in theory, provide an exponential advantage over classical methods, there are still significant resource requirements that need to be overcome. In particular, the number of qubits, the circuit depth, and the number of measurements exceed the capabilities of existing technologies and present active research topics.

VQE is typically performed in the framework of Second Quantization. In this thesis, we explore the possibility of formulating the algorithm in First Quantization. In particular, we aim to answer the following questions: *i*) can every step of the algorithm be implemented efficiently in First Quantization? and *ii*) what are the advantages and disadvantages in performance compared to Second Quantization approaches?

To answer these questions, we investigate the performance and resource requirements of the fermion-to-qubit encoding, the antisymmetrization, the measurement cost, and the Ansatz choice. We find that the first quantized encoding leads to a significant reduction in the number of qubits. However, this typically comes at the cost of increasing the number of measurements. To solve this problem, we identify an approach based on plane waves as a promising method for significantly reducing the number of measurements without changing the overall circuit depth. However, this comes at the cost of complicating the initial state preparation. Furthermore, we show that many typically used Ansätze are inefficient in First Quantization. We identify the qubit-ADAPT-VQE Ansatz as an Ansatz that performs at least as good as its counterpart in Second Quantization.

Finally, we address the problem of combining the Ansatz circuit and the antisymmetrization. We show that this is a non-trivial task that requires adjustments in the antisymmetrization methods that are typically used. We demonstrate this by applying this method to the qubit-ADAPT-VQE Ansatz and performing numerical simulations to determine the probability of successfully preparing antisymmetrized Ansatz states. We find that this can be done efficiently, in particular when describing a small number of electrons with high accuracy.

Our contributions provide new insights into the feasibility and performance of a First Quantization approach to VQE, which could have implications for the development of more efficient algorithms for quantum chemistry simulations.

Zusammenfassung

Die Durchführung von Quantenchemie-Simulationen ist eine der wichtigsten potenziellen Anwendungen von Quantencomputern in der nahen Zukunft. Einer der prominentesten Algorithmen hierfür ist der Variational Quantum Eigensolver (VQE), der es uns ermöglicht, Grundzustände von molekularen Hamiltonians zu berechnen. Obwohl VQE in der Theorie gegenüber klassischen Methoden einen exponentiellen Vorteil bietet, gibt es immer noch signifikante Ressourcenanforderungen, die überwunden werden müssen. Insbesondere die Anzahl der Qubits, die Schaltungstiefe und die Anzahl der Messungen übersteigen die Fähigkeiten bestehender Technologien und stellen aktive Forschungsthemen dar.

VQE wird in der Regel im Rahmen der zweiten Quantisierung durchgeführt. In dieser Arbeit untersuchen wir die Möglichkeit, den Algorithmus in der ersten Quantisierung zu formulieren. Insbesondere möchten wir folgende Fragen beantworten: i) Kann jeder Schritt des Algorithmus in der ersten Quantisierung effizient implementiert werden? und ii) Was sind die Vor- und Nachteile im Vergleich zu Ansätzen in der zweiten Quantisierung? Um diese Fragen zu beantworten, untersuchen wir die Performance und Ressourcenanforderungen der Fermion-zu-Qubit-Codierung, der Antisymmetrisierung, der Messkosten und der Ansatzwahl. Wir stellen fest, dass die erste quantisierte Codierung zu einer signifikanten Reduktion der Anzahl von Qubits führt. Dies geschieht jedoch in der Regel auf Kosten einer Erhöhung der Anzahl von Messungen. Um dieses Problem zu lösen, identifizieren wir einen Ansatz basierend auf ebenen Wellen als vielversprechende Methode zur signifikanten Reduktion der Anzahl von Messungen ohne Änderung der Gesamtschaltungstiefe. Dies geht jedoch auf Kosten einer Komplikation der Anfangszustandspräparation. Außerdem zeigen wir, dass viele üblicherweise verwendete Ansätze in der ersten Quantisierung ineffizient sind. Wir identifizieren den Qubit-ADAPT-VQE-Ansatz als einen Ansatz, der mindestens genauso gut abschneidet wie sein Gegenstück in der zweiten Quantisierung.

Schließlich behandeln wir das Problem der Kombination des Ansatzes und der Antisymmetrierung. Wir zeigen, dass dies eine nicht triviale Aufgabe ist, die Anpassungen in den Antisymmetrierungsmethoden erfordert, die normalerweise verwendet werden. Wir demonstrieren dies, indem wir diese Methode auf den Qubit-ADAPT-VQE-Ansatz anwenden und numerische Simulationen durchführen, um die Wahrscheinlichkeit der erfolgreichen Vorbereitung antisymmetrisierter Ansatzzustände zu bestimmen. Wir stellen fest, dass dies effizient geschehen kann, insbesondere bei der Beschreibung einer kleinen Anzahl von Elektronen mit hoher Genauigkeit.

Unsere Beiträge bieten neue Einblicke in die Durchführbarkeit und Leistung eines Ansatzes in der ersten Quantisierung für VQE, was Auswirkungen auf die Entwicklung effizienterer Algorithmen für Quantenchemie-Simulationen haben könnte.

Contents

1 Introduction

In recent years, quantum computing has experienced a rapid increase in popularity and is regularly portrayed as the next major technological revolution [7]. Public confidence in the promises of this new technology has grown so much that investment in quantum computing surpassed $3 billion by 2021 [42]. In fact, some sources even predict the global quantum computing market to grow to as much as $125 billion by 2030 [50]. This excitement and optimism is driven by the many real-life applications of quantum computers, some of which may have a considerable impact on our lives. Probably the most prominent example of a real-world application of quantum computing is Shor's algorithm for factoring large numbers into prime factors [54]. Given that factoring large numbers is believed to be impossible to perform efficiently on classical computers, it is used as the main building block of the popular RSA encryption method. A full-scale quantum computer would be able to solve the factoring problem and read any RSA encrypted data, leading to yet unimaginable consequences in an increasingly digital world. On the other hand, quantum computing may enable fundamentally secure communication and create unbreakable encryption methods [12]. Other regularly mentioned uses include financial modelling, traffic flow optimization, and weather forecasting [16].

As impressive as these applications sound, they share a common problem: existing quantum computers are not yet accurate nor large enough to be useful. Today's largest working quantum computers are able to control a few hundred qubits [28], while controlling a few thousand qubits is predicted to be possible in the coming years [29]. Yet, breaking the widely used 2,048 bit RSA encryption, for instance, would require approximately 20 million qubits [23]. This shows that even our best technologies are still very far from systems that are useful in real-life applications. In fact, some people argue that we might never achieve this goal, either because of the tremendous technological challenges that would need to be overcome or because of certain fundamental principles of quantum physics itself [1].

Historically, the original motivation behind quantum computers was a different one. The idea has been around since 1981, when Richard Feynman published his famous paper "Simulating Physics with Computers" [21], in which he argued that quantum mechanical systems cannot be simulated efficiently on classical devices. A natural consequence of this finding was to build a computer that is itself based on a well controlled quantum system, thereby integrating quantum mechanics into the computational process. Manipulating this system would then make it possible to simulate other quantum systems that are much harder to generate and control in a lab. If successful, quantum simulation could play an important role in fundamental research, drug discovery [15], efficient production of fertilizer [49], and the development of novel materials for battery and sustainable energy production technologies [5].

At the heart of these applications is the field of quantum chemistry, i.e. the accurate prediction of chemical properties of molecules. More specifically, molecules are treated as quantum mechanical systems and properties such as ground states and energy levels are obtained by applying the Schrödinger equation. Several methods have been developed to solve the quantum chemistry problem by performing simulations on classical computers, for instance Hartree-Fock theory [31], Density Functional Theory [45], and Densiy-Matrix-Renormalization-Group (DMRG) [52]. However, these methods are only approximations to the real solutions. Exact methods, such as the Full Configuration Interaction (FCI) [51] provide more accurate results, however, the classical resources to perform FCI computations scale exponentially in the system size and precision, such that even the best classical supercomputers quickly reach their limit. A quantum computer, on the other hand, does not suffer from this problem and could, in theory, efficiently perform simulations that are impossible on classical machines [10].

Admittedly, we are still far from simulating large molecules on a real quantum computer, but there is hope: while Shor's algorithm requires large numbers of error corrected qubits, some quantum chemistry simulations might be possible without error correction. Instead of waiting for the advent of fault tolerance, people are now starting to embrace current and near-term technologies. Recently termed NISQ (Noisy Intermediate Scale Quantum Computing) [47], this stage of development in quantum computing is defined by the acceptance of certain error rates, restrictions in size, and finite coherence times. Although no significant advantage has yet been achieved in real experiments, several NISQ algorithms have been proposed with potential applications ranging from chemistry and machine-learning to financial portfolio optimization [14].

Probably one of the most important proposed NISQ algorithms, serving as the main topic of this thesis, are Variational Quantum Algorithms (VQA). VQAs are particularly relevant in the context of quantum chemistry, since they can be used to obtain ground states of molecular Hamiltonians exponentially more efficiently than classical computers. Variational quantum algorithms were first introduced by Peruzzo et al. [46] in 2014. They are a class of algorithms whose aim is to minimize an objective function $\mathcal{O}$. If the objective function is chosen to be the energy of a quantum state, the resulting algorithm is called Variational Quantum Eigensolver (VQE). In this case, the goal is to find the ground state of a quantum mechanical system described by a Hamiltonian H. Apart from the choice of objective function, the general structure of all VQAs is very similar. We will therefore only consider VQE here and refer to [46] for a broader and more thorough review. The VQE algorithm consists of three steps: i) prepare a parameterized initial state; ii) measure the energy; iii) embed these two steps in a classical optimization routine to find the state with the lowest energy. Each step constitutes an active area of research in which the details depend highly on the system considered and on available quantum computing architecture.

First, an initial state is prepared by applying a quantum circuit that is determined by a set of classical parameters θ. This collection of parameterized unitary one- and two-qubit operations is typically known as the *Ansatz*. Next, we estimate the expectation value $\langle H \rangle$ with respect to the current state. To do this, we translate the operator H from an operator in Fock space to an operator in qubit space using, for example, a Jordan-Wigner transformation and measure each term individually. In practice, this step amounts to measuring a set of Pauli strings, i.e., tensor products of Pauli matrices. A detailed description of this step is provided in section 3.2. Once the energy $E(\theta)$ is obtained, a classical optimization method is applied to adjust the parameters θ and repeat the procedure until the energy is minimized. The final set of parameters θ_0 then represent an approximation of the real ground state of the Hamiltonian.

From the structure of this method it becomes clear why VQE attracts attention as a promising NISQ algorithm. Relying on a classical computer for the parameter optimization enables much shallower circuits as compared to other quantum algorithms. In fact, the only task that needs to be performed on a quantum computer is the state preparation, immediately followed by a measurement, taking advantage of the relatively short coherence times. Despite this promising outlook, there are still major obstacles in the way of a useful implementation of VQE on a real device. In particular, the number of measurements needed to accurately estimate the expectation values of the Hamiltonian is considered to be one of the biggest limiting factors [57]. One might argue, however, that the number of qubits in the circuit and the number of gates in the Ansatz represent challenges that are just as serious. This is because the number of measurements only determines the time of the computation, whereas the depth and number of qubits are determined by the limits of exciting technologies.

The main application of VQE is to perform quantum chemistry simulations, i.e. to find the ground state corresponding to some atom, molecule or crystal. More specifically, we typically treat atomic nuclei as fixed and only consider the electronic wavefunctions. Since they are fermions, electrons are fundamentally different from qubits and it is not obvious how to encode their quantum state on a quantum computer. In 1928, Jordan and Wigner introduced a way of mapping fermions to qubits [33], which is still widely used in VQE to solve this problem. Other methods include the Bravyi-Kitaev encoding [17], and the Parity encoding. The choice of encoding ultimately influences the performance of the computation and optimizing this step is an active area of research. All typically used encodings have one thing in common, however. They are carried out using the formalism of Second Quantization. In other words, the antisymmetry resulting from the Pauli exclusion principle is enforced within the anti-commutation relations of the creation and annihilation operators. Although evidently a successful method in the context of VQE, this is not the only possible way. In particular, one might enforce antisymmetry through the wavefunction directly, traditionally referred to as First Quantization. To our know-

ledge there has not yet been any research into the potential of VQE in First Quantization. In fact, if mentioned at all, this approach is usually discarded as "too costly for the NISQ era" [57]. Intuitively, there is some reason to believe that the two approaches lead to different outcomes, but it is not obvious whether one is superior in general. Encodings in Second Quantization naturally require a number of qubits that is equal to the number of electrons m, as well as the number of electronic orbitals N. In First Quantization, however, the number of qubits required is $m \log N$ (see section 4.1 for a detailed explanation). This approach might thus provide an advantage in scenarios where we are interested in describing small molecules (low m) with high accuracy (large N). The aim of this thesis is to explore different approaches to VQE in First Quantization and determine their advantages and disadvantages. More specifically, we examine how the number of qubits, measurements and quantum gates scale as a function of the number of orbitals and electrons used in the computation.

We begin in chapter 2 with an overview of the necessary background information about quantum computing and quantum chemistry. Chapter 3 is dedicated to an in-depth review of common practices and results of the Second Quantized approach to VQE. We discuss several established encoding and Ansatz choices and outline possible strategies of improving the gate depth and measurement cost. Chapter 4 explores potential approaches to VQE in First Quantization. We begin by establishing the first quantized Hamiltonian and describe how to encode the electronic wavefunction on a quantum register. Next, we discuss the antisymmetrization of the wavefunction and examine how to implement this method efficiently on a quantum computer. In section 4.3, we introduce a measurement strategy based on plane waves, demonstrating both advantages and disadvantages compared to the Second Quantized case. In section 4.4, we propose a full version of VQE in First Quantization that combines the qubit-ADAPT-VQE Ansatz [55] discussed in section 3.3.2 and methods from the antisymmetrization procedure discussed in section 4.2. Finally, chapter 5 is dedicated to numerical simulations. In order to test the viability of the proposed Ansatz, we implement the relevant circuits using the python package `QISKIT` and investigate the behaviour in terms of the number of electrons and basis functions.

2 Theoretical Background

2.1 Quantum Computing

The theory of modern quantum mechanics was developed in the early 1920s, following decades of increasing evidence that traditional, "classical", physics is not sufficient to accurately describe microscopic phenomena. Consequently, quantum mechanics became an indispensable tool in technology and science. The study of quantum information was introduced in the 1960s when physicists raised questions such as: "Does quantum physics allow the transmission of information faster than the speed of light?" and "Can quantum information be copied?" [22]. Both questions turned out to be related by the *No-Cloning Theorem* [58] and suggested that quantum information is fundamentally different from classical information. At the same time, physicists strove to find methods to completely control quantum systems in the lab. Achieving this would mean to experimentally access an entirely new branch of nature, comparable to the invention of radio astronomy in the 1930s. Both concepts of quantum information and controlled quantum systems were then combined into the notion of quantum computers.
Today, there are several models that can be used to formally describe quantum computations. The most common one is the circuit model [43], which may be thought of as the quantum analogue of classical circuits. In this chapter, we will give a summary of the circuit model and introduce the individual building blocks that are necessary to build quantum algorithms. In particular, we will define those quantum gates that are used in our implementation of VQE in section 4 and introduce measurements of Pauli string expectation values.

2.1.1 Qubits

Single Qubits

Quantum bits, or qubits for short, are the fundamental building blocks of quantum computers and represent the quantum analogue of classical bits. A bit is typically portrayed as the smallest unit of information and can hold either the value **1** ('True') or **0** ('False'). When multiple bits are connected through logic gates we can perform computations from simple addition of two numbers to writing a master thesis on LaTeX. In close analogy, a qubit should be a quantum mechanical system that, when measured, returns either **1** or **0**. Such a system is referred to as a two-level system and fully characterized by the two basis states $|0\rangle$ and $|1\rangle$ of a two-dimensional complex Hilbert space. It turns out that using these qubits we can perform any computation that we can also perform using classical bits [43]. The question now is whether using qubits provides any advantage over using classical bits. As a first indication of this we observe that a qubit can occupy more states that a classical bit, namely, prior to any measurement it can exist in a superposition of

both basis states, i.e.

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle\ . \tag{1}$$

The complex amplitudes α and β determine the probabilities of the possible measurement outcomes $\mathrm{P}_0 = |\alpha|^2$ and $\mathrm{P}_1 = |\beta|^2$ and are thus chosen to satisfy $|\alpha|^2 + |\beta|^2 = 1$. This normalization together with invariance under multiplication by a complex phase implies the Bloch representation

$$|\psi\rangle = \cos(\theta/2)\,|0\rangle + e^{i\varphi}\sin(\theta/2)\,|1\rangle\ , \tag{2}$$

using only two real parameters θ and φ. In this context, we can think of the state of a qubit as a point on the surface of a three-dimensional sphere, also referred to as the Bloch sphere. Due to the unitary nature of quantum mechanics, any transformation on a single qubit can therefore be realized as a simple rotation on the Bloch sphere.

Multiple Qubits

Despite exhibiting interesting properties, single qubits are not sufficient for universal computation. One could argue that the essential effects that make quantum computing interesting only occur when multiple qubits are connected in a non-trivial way. The state space of n qubits is given by the tensor product of the individual qubit spaces. Formally, if a single qubit is described by the Hilbert space $\mathcal{H}$, the space of n qubits is given by

$$\mathcal{H}_n = \overbrace{\mathcal{H} \otimes ... \otimes \mathcal{H}}^{n \text{ times}}\ . \tag{3}$$

Whereas the single qubit space is spanned by the two orthogonal vectors $|0\rangle$ and $|1\rangle$, the n qubit space is now spanned by 2^n basis vectors, represented by all possible binary strings of length n. This already illuminates some of the difficulties of handling quantum states on a classical computer. Simply storing the necessary information to completely describe the state of n particles requires 2^n complex numbers.

The tensor product nature of the n qubit Hilbert space has another important consequence: it allows states to be entangled. Just as a single qubit can be in a superposition of two different states, multiple qubits can be in a superposition of two different multi-qubit states. As an example, consider the Bell state

$$|\Psi\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}}\ . \tag{4}$$

The two qubits are now correlated in a way that cannot be replicated in classical physics. While measurements on each individual qubit are completely random, they always return the same result, regardless of which qubit was measured first. In other words, if a measurement on the first qubit results in the outcome **0**, the state of the second qubit will instantaneously collapse to the state $|0\rangle$. Whether this collapse constitutes a real, physical change in the particles is a highly debated question and was first raised by Einstein, Rosen, and Podolsky in 1935 [20]. Luckily, we don't need to answer this question to perform interesting quantum computations. In general, a state $|\psi\rangle \in \mathcal{H}_{\mathrm{AB}}$ is called *entangled*, if it cannot be written as a simple product state $|\psi\rangle_{\mathrm{A}} \otimes |\psi\rangle_{\mathrm{B}}$.

Mixed States

Aside from the purely quantum mechanical effects, qubits can also exhibit regular, classical uncertainty. In particular, it is possible to define probability distributions over several different states. Moreover, this probability distribution is itself a quantum state and typically referred to as a *density matrix*. In the special case when the system is in a single state $|\psi\rangle$ with certainty, we define the density operator $\rho = |\psi\rangle\langle\psi|$ and call it a *pure state*. If the probability distribution is non-trivial, we refer to it as a *mixed state* given by

$$\rho = \sum_i p_i |\psi_i\rangle\langle\psi_i| \ , \tag{5}$$

with probabilities $p_i \in [0,1]$, such that $\sum_i p_i = 1$. In this case the system is in the pure state $|\psi\rangle$ with probability p_i. More abstractly, a density operator can be defined as a positive semi-definite, Hermitian operator of trace one. An important application of this is the detection of bipartite entanglement in pure states. To see this, consider two subsystems $\mathcal{H}_{\mathrm{A}}$ and $\mathcal{H}_{\mathrm{B}}$ of a larger space $\mathcal{H}_{\mathrm{AB}}$. A pure state $|\psi\rangle_{\mathrm{AB}} \in \mathcal{H}_{\mathrm{AB}}$ is entangled if and only if the reduced states ρ_{A} and ρ_{B} are mixed. Here, the reduced states are obtained by tracing out the respective other half of the system, i.e. $\rho_{\mathrm{A}} = \mathrm{Tr}_{\mathrm{B}}(|\psi_{\mathrm{AB}}\rangle\langle\psi_{\mathrm{AB}}|)$.

2.1.2 Quantum Gates

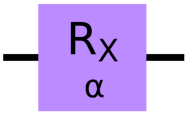
Having defined the qubit as a quantum analogue of the bit, it is now necessary to connect them in circuits to perform computations. Whereas classical circuits consist of bits connected through logic gates, quantum circuits are build by connecting qubits and performing transformations in the form of quantum gates. There is an important difference between the two models, however. Namely, classical gates can often be irreversible, while quantum gates *have* to be reversible. This is due to the fact that any valid transformation between (pure) quantum states must be unitary. Consequently, quantum gates are described by unitary operators.
The first step of building quantum circuits is to choose a convenient representation of qubit states and operators. In particular, we represent the ket and bra version of the state (1) as the vectors

$$|\psi\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \qquad \text{and} \qquad \langle\psi| = \begin{pmatrix} \alpha^* & \beta^* \end{pmatrix}, \tag{6}$$

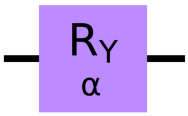
respectively. States of multiple qubits are then easily constructed by applying the Kronecker product between the vectors, and inner products are evaluated using a simple matrix product. Similarly, unitary transformation between states become unitary matrices between vectors and are called quantum gates. The dimension of these matrices quickly grows as $2^n \times 2^n$ for n-qubit states, again confirming the notion that qubits are difficult to simulate classically. In the following, we will introduce the most important gates that are regularly used to construct quantum circuits and algorithms.

Single Qubit Gates

As mentioned in Sec. 2.1.1, pure single qubit states can naturally be represented as vectors on the three-dimensional Bloch sphere. Unitary quantum gates can therefore be described by rotations around the x, y, and z axis within this picture. In fact, all single qubit transformations can be spanned by combining the three rotation gates R_x, R_y, R_z in equations (7) - (9) and the phase gate P [43] in equation (10). Each rotation gate is generated by one of the Pauli matrices (11) - (13).
In the circuit model, qubits are depicted as horizontal lines and evolve with time from left to right. Quantum gates are depicted as boxes on top of the qubit wires and are labeled with their name and possible parameters. If not mentioned otherwise, all qubits are initialized in the $|0\rangle$ state.



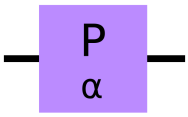
$$\mathsf{R}_x(\alpha) = e^{-i\alpha/2X} = \begin{pmatrix} \cos\frac{\alpha}{2} & i\sin\frac{\alpha}{2} \\ i\sin\frac{\alpha}{2} & \cos\frac{\alpha}{2} \end{pmatrix} \tag{7}$$



$$\mathsf{R}_y(\alpha) = e^{-i\alpha/2Y} = \begin{pmatrix} \cos\frac{\alpha}{2} & -\sin\frac{\alpha}{2} \\ \sin\frac{\alpha}{2} & \cos\frac{\alpha}{2} \end{pmatrix} \tag{8}$$

$$\mathsf{R}_z(\alpha) = e^{-i\alpha/2Z} = \begin{pmatrix} e^{-i\frac{\alpha}{2}} & 0 \\ 0 & e^{i\frac{\alpha}{2}} \end{pmatrix} \tag{9}$$

The phase gate P only differs from the rotation gate R_z by a complex phase factor. If we are only interested in a single qubit, such a phase factor can be ignored, but it becomes relevant once multiple qubits are combined.



$$\mathsf{P} = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\alpha} \end{pmatrix} \tag{10}$$

Even though these gates are sufficient, it is common to define specific instances that are very frequently used in practice. Most importantly, the X, Y, and Z gates are defined as rotations by $\alpha = \pi$ around their respective axes and coincide withe the Pauli matrices.

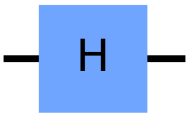
$$\mathsf{X} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \tag{11}$$

$$\mathsf{Y} = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \tag{12}$$

$$\mathsf{Z} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \tag{13}$$

Particularly important for many quantum algorithms is the Hadamard gate. It is equivalent to a rotation around the x-axis by 90°, followed by a rotation around the y-axis by 180°. It is typically used to create equal superposition states, since for example, $H|0\rangle = (|0\rangle + |1\rangle)/\sqrt{2}$. By applying it to each qubit in the product state $|0\rangle^{\otimes n}$ we can

generate an equal superposition of all possible binary strings of length n. This is a common starting point for many quantum algorithms, as we can then evaluate functions on each element of the superposition, i.e. perform different operations in parallel.

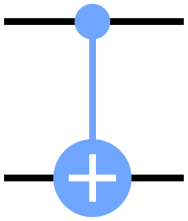


$$\mathsf{H} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \tag{14}$$

Controlled Quantum Gates

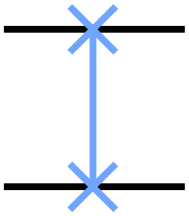
Single-qubit rotations allow us to construct arbitrary product states, but by themselves they are not able to generate entanglement. Evidently, we need gates that act on multiple qubits. Most commonly, these interactions are implemented as controlled gates, where we apply a transformation to one qubit, conditioned on the state of another.

The prototypical example is the CNOT-gate, or controlled-NOT gate. The action of the gate is to apply an X-gate to the *target* qubit if and only if the *control* qubit is in the state $|1\rangle$. The control and target qubits are depicted as small and large circles, respectively, connected by a straight vertical line:



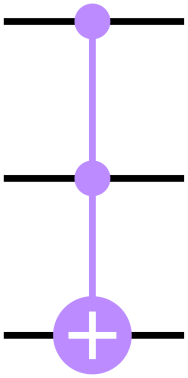
$$\mathsf{CNOT}_{01} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{15}$$

Direct interactions between quantum mechanical particles can only be achieved if they are close together. In order to implement two-qubit gates on qubits that are far apart in the circuit, it is therefore necessary to move them around. This is achieved by the SWAP-gate, which simply interchanges two qubits. By successively applying the SWAP-gate to adjacent wires, we can then move qubits to any desired location in the circuit.



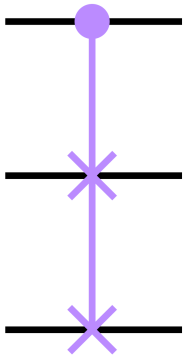
$$\mathsf{SWAP}_{01} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{16}$$

Of course, we are not limited to only two-qubit gates and can easily define gates between three or more qubits. Typically, those gates simply increase the number of control qubits. A famous example of a three-qubit gate is the TOFFOLI-gate, which is equivalent to a CNOT-gate with one additional control qubit.



$$\mathsf{TOFFOLI}_{012} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \tag{17}$$

Particularly relevant for quantum chemistry simulations is the CSWAP-gate in equation (18), which swaps two qubits conditioned on the state of a third one. As we will explore in section 4.2, working in the framework of first quantization requires us to construct antisymmetric quantum states of multiple electrons. In that context, the CSWAP-gate plays an important role in generating superposition states of different orderings of a particular product state.



$$\mathsf{CSWAP}_{012} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{18}$$

Universality

Having introduced a collection of quantum gates, one might now ask how many different ones we actually need. In other words, is there a small set of gates from which any n-qubit unitary can be constructed? In analogy to the classical case, we call such sets *universal*. It turns out that this can be achieved in several different ways. In fact, assuming we can generate any single qubit transformation, simply adding the CNOT-gate enables us to perform any computation on n qubits exactly. In particular, this implies that it is not actually necessary to directly implement gates on three or more qubits. Following the reasoning above, we might choose $\{\mathrm{R}_x, \mathrm{R}_y, \mathrm{R}_z, \mathrm{P}, \mathrm{CNOT}\}$ as universal gate set [43]. It is possible to relax the notion of universality, such that we're only interested in approximating n-qubit unitaries. In that case we can choose gate sets such as $\{\mathrm{TOFFOLI}, \mathrm{H}\}$ or $\{\mathrm{CNOT}, \mathrm{H}, \mathrm{S}, \mathrm{T}\}$, where $\mathrm{T} = P(\pi/4)$ and $\mathrm{S} = T^2$ [3].

2.1.3 Measurements

After applying a sequence of gates, the qubit state stores information that we would like to extract. In the case of classical computation, this step is trivial. One can simply measure the voltage on certain output wires without disturbing the state of the system. In quantum circuits, however, it is not possible to directly access the state. Any measurement on one qubit leads to an instantaneous collapse of the wavefunction and thus changes the outcome of measurements on the other qubits. As a consequence, quantum algorithms can only work probabilistically and require many repetitions to generate accurate results. By default, it is only possible to measure each individual qubit in the Pauli z-basis, also referred to as the *computational basis*. However, by applying single-qubit rotation gates we can change basis and perform measurements in the x and y-basis as well. Computational basis measurements are effectively measurements of the observable Z. In order to measure an observable A, we thus need to find a unitary transformation U, such that

$$Z = U^\dagger A U \, . \tag{19}$$

By this reasoning, x-basis measurements require the application of a Hadamard gate and y-basis measurements an $\mathrm{R}_x(\pi/2)$-gate. In the circuit model we depict measurements as black boxes with arrows indicating the classical register where the measurement outcome is stored.

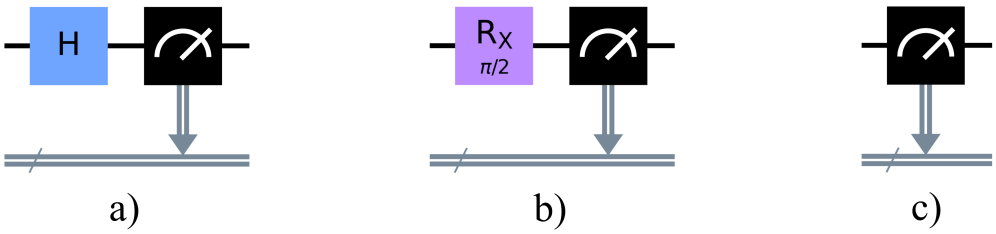


FIGURE 1: Single qubit measurements in a) the x-basis, b) the y-basis, and c) the z-basis

This procedure generalizes easily to the case of multiple qubits, allowing us to measure arbitrary Pauli strings. In particular a Pauli string $\mathbf{P} = \mathbf{P_1} \otimes ... \otimes \mathbf{P_n}$ requires the unitary $U_1 \otimes ... \otimes U_n$, where each U_i is obtained by solving $Z = U_i^\dagger P_i U_i$.

2.2 Noisy Intermediate Scale Quantum Computing (NISQ)

The attempt to build a real quantum computer in the lab may seem paradoxical at first. On the one hand, the qubits need to be perfectly isolated from the environment and shielded from radiation, magnetic fields, air molecules, etc. On the other hand, we need to be able to precisely control and change the state of the qubits, perform measurements and thus extract information. This dichotomy is arguably the main challenge that comes with building a quantum computer.

The property of quantum systems to interact with the environment is also referred to as *decoherence* and leads to a loss of quantum behaviour over time. As a consequence, real quantum computers are limited by their finite coherence time, i.e., the available time in which the qubits are well-isolated and all computations have to be carried out. The ability to overcome this challenge and to control qubits to a degree necessary for arbitrary quantum algorithms is referred to as *fault tolerance*. There is a consensus that if fault tolerance can be achieved, it will happen through the use of quantum error correction, a method that was first proposed in 1995 by Peter Shor [53]. His result was further developed into the *threshold theorem* by Aharonov et al. [4], Knill et al. [37] and Kitaev [35] independently. The theorem states that if the physical error rate can be kept below a certain threshold, then it is possible to suppress the logical error rate to arbitrary low levels by applying error correction schemes. These schemes usually require several physical qubits for each logical qubit and thus substantially increase the total numbers of qubits. In order to perform useful computations, this could easily exceed millions of qubits and therefore does not belong into the reality of near-term technologies. The term NISQ (Noisy Intermediate Scale Quantum Computing) was coined by John Preskill in 2018 [47] and encompasses the quantum computing hardware of today and the near future. In particular, we are interested in finding useful algorithms and applications for devices with a certain level of noise above the error correction threshold and limitations in the number of qubits.

2.3 Quantum Chemistry Simulations

Having covered the fundamentals of quantum computing, we now turn to the main application that is central to this thesis: the simulation of molecular systems. A detailed review of this field can be found in [10]. The aim of this chapter is to give a mathematical description of molecules as quantum mechanical systems and explore how methods on both classical and quantum computers may be used to obtain solutions to this problem.

2.3.1 Electronic Structure Problem

Within the theory of quantum mechanics, molecules are defined as a collection of atoms and electrons whose quantum state is a solution to the time-independent, non-relativistic Schrödinger equation

$$H\left|\psi\right\rangle = E\left|\psi\right\rangle \ . \tag{20}$$

Solutions to this eigenvalue equation are given by a set of states $|\psi\rangle$ representing the energy levels of a molecule with corresponding energies E. The molecular Hamiltonian H contains all relevant information about the electrons and nuclei and is defined as

$$\begin{aligned} H = &\overbrace{-\frac{\hbar^2}{2m_e}\sum_{i=1}^{m}\nabla_i^2}^{T_e} \overbrace{-\sum_{k=1}^{M}\frac{\hbar^2}{2m_k}\nabla_i^2}^{T_{\text{nuc}}} + \overbrace{\frac{1}{4\pi\epsilon_0}\frac{1}{2}\sum_{i\neq j}^{m}\frac{e^2}{|\boldsymbol{r}_i-\boldsymbol{r}_j|}}^{V_{e,e}} \\ &\underbrace{-\frac{1}{4\pi\epsilon_0}\sum_{k=1}^{M}\sum_{i=1}^{m}\frac{Z_k e^2}{|\boldsymbol{r}_i-\boldsymbol{R}_k|}}_{V_{e,\text{nuc}}} + \underbrace{\frac{1}{4\pi\epsilon_0}\frac{1}{2}\sum_{k\neq l}^{M}\frac{Z_k Z_l e^2}{|\boldsymbol{R}_k-\boldsymbol{R}_l|}}_{V_{\text{nuc,nuc}}} \ . \end{aligned} \tag{21}$$

Here, T_e and T_{nuc} describe the kinetic energy of a set of m electrons and M nuclei and $V_{e,e}$, $V_{e,\text{nuc}}$, and $V_{\text{nuc,nuc}}$ describe the Coulomb potentials between electron-electron, electron-nucleus, and nucleus-nucleus, respectively. m_e and m_k are the masses of electrons and nuclei, $\mathbf{r_i}$ and $\mathbf{R_k}$ their positions, and Z_k the atomic number of the nuclei. Due to the significantly higher mass of the nuclei compared to the electrons, it is common to work in the Born-Oppenheimer approximation. In this approximation, we assume that the electronic and nuclear motions can be separated and solved independently, resulting in a considerable reduction in complexity but only small deviations from the true solution. The state $|\psi\rangle$ may then be written as a product $|\psi_e\rangle \otimes |\psi_{\text{nuc}}\rangle$ consisting of the electronic and nuclear wavefunctions. In the following, we will exclusively focus on the electronic wavefunction with a constant background of nuclear charges, also referred to as the *electronic structure problem*. We may therefore write the molecular Hamiltonian as

$$H_{\mathrm{BO}} = T_e + V_{e,\mathrm{nuc}} + V_{e,e}\,. \tag{22}$$

One of the main goals of electronic structure computations is to find ground states of the Hamiltonian (22). More specifically, we are interested in the state $|\psi_0\rangle$ with the lowest energy eigenvalue E_0, also called the ground state energy.
There are many reasons for the importance of electronic structure calculations in science and technology. For instance, determining the molecular ground state and excitation energies enables us to predict various electrical, mechanical, and magnetic properties of materials. Moreover, the results can be used to understand and simulate chemical reactions. This is particularly important in cases where the molecules in question are difficult or expensive to study in the lab. One notable example of a process that we do not understand on a chemical level is nitrogen fixation by the enzyme nitrogenase, which is able to transform dinitrogen into ammonia molecules under ambient conditions. Ammonia is an indispensable ingredient in the production of artificial fertilizer and with today's technologies only achievable by means of the very energy intensive Haber-Bosch catalyst. Unfortunately, the nitrogenase molecule is far too complex to be solved using classical methods. A quantum computer, on the other hand, could in principle solve the electronic structure problem efficiently and thus have a significant impact on the availability of cheap, green fertilizer [49].

2.3.2 Discretization and Basis Choice

Regardless of whether we use a classical or quantum computer, it is not obvious how to encode the electronic structure problem in a way that can be utilized by a computer. The difficulty here is that the solutions to equation (20) are in general elements of an infinite dimensional Hilbert space. The first step in resolving this problem is to define a complete set of single-electron basis functions $\{|\phi_p\rangle\}$ from which the electronic wavefunctions are built. Technically, each basis function is divided into its spatial and spin components, but we will omit this distinction here for clarity. Throughout the thesis we will use the terms *basis function*, *orbital*, and *fermionic mode* interchangeably. Since electrons are fermions, it is necessary for the wavefunction to be antisymmetric under exchange of any two electrons:

$$|\phi_1 \ldots \phi_i \ldots \phi_j \ldots \phi_m\rangle = -\,|\phi_1 \ldots \phi_j \ldots \phi_i \ldots \phi_m\rangle\ . \tag{23}$$

It is therefore not possible to obtain a valid many-electron wavefunction by simply combining basis functions in a tensor product $|\phi_1\rangle \otimes \ldots \otimes |\phi_m\rangle$. Instead, one typically uses a *Slater determinant*, defined as a linear combination of all possible permutations σ of the

m electrons with a corresponding phase factor.

$$|\phi_1 \dots \phi_m\rangle := \frac{1}{\sqrt{m!}} \sum_{\sigma \in S_m} (-1)^\sigma \left|\phi_{\sigma(1)}\right\rangle \otimes \dots \otimes \left|\phi_{\sigma(m)}\right\rangle \tag{24}$$

The true electronic wavefunction is then given by a linear combination of all such Slater determinants. So far, we have only reformulated the problem and thus the state space is still infinite dimensional. However, we can now to choose a finite subset of the single-electron basis functions and thereby introduce an approximation. However, it can be shown that the discretization error scales as $\mathcal{O}(N^{-1})$, where N is the number of basis functions [8]. If we project the Hamiltonian (22) onto the space spanned by the set of basis functions $\{|\phi_p\rangle\}_{p=1}^N$ we can rewrite it in terms of one- and two-electron matrix elements [57] as

$$H = \sum_{i=1}^{m} \sum_{p,q=1}^{N} h_{pq} \left|\phi_p\right\rangle_i \left\langle\phi_q\right|_i + \frac{1}{2} \sum_{i \neq j}^{m} \sum_{p,q,r,s=1}^{N} h_{pqrs} \left|\phi_p\right\rangle_i \left|\phi_r\right\rangle_j \left\langle\phi_q\right|_i \left\langle\phi_s\right|_j . \tag{25}$$

The matrix elements h_{pq} can be obtained by evaluating the kinetic term and potential interaction between electrons and nuclei and the matrix elements h_{pqrs} are obtained by evaluating the interaction integrals resulting from the electron-electron potential.

$$h_{pq} = \langle\phi_p| T_e + V_{e,\text{nuc}} |\phi_q\rangle = \int \mathrm{d}\boldsymbol{r}\, \phi_p^*(\boldsymbol{r}) \left(-\frac{\hbar^2}{2m_e} \nabla^2 - \sum_{l=1}^{M} \frac{e^2}{4\pi\epsilon_0} \frac{Z_l}{|\boldsymbol{r} - \boldsymbol{R}_l|} \right) \phi_q(\boldsymbol{r}) \tag{26}$$

$$h_{pqrs} = \langle\phi_p\phi_r| V_{e,e} |\phi_q\phi_s\rangle = \frac{e^2}{4\pi\epsilon_0} \int \mathrm{d}\boldsymbol{r}_1 \mathrm{d}\boldsymbol{r}_2 \frac{\phi_p^*(\boldsymbol{r}_1)\, \phi_r^*(\boldsymbol{r}_2)\, \phi_q(\boldsymbol{r}_2)\, \phi_s(\boldsymbol{r}_1)}{|\boldsymbol{r}_1 - \boldsymbol{r}_2|} \tag{27}$$

The concrete choice of basis functions depends on the problem at hand and on the technique that is used to solve it. *Atomic Orbitals* are typically used when describing single molecules or systems that are localized in space. They are further distinguished into *Gaussian-type* orbitals (GTO) and *Slater-type* orbitals (STO), which differ in the radial part $R(r)$ of the wavefunction. Both sets of functions are localized around the positions of the nuclei:

$$\text{Slater:} \quad R_l(r) \propto r^l e^{-\alpha r} \qquad\qquad \text{Gaussian:} \quad R_l(r) \propto r^l e^{-\alpha r^2} \tag{28}$$

The advantage of Gaussian-type basis sets is that the integrals (26) and (27) are far easier

to compute compared to Slater-type basis sets. The third common choice is the *plane wave* basis that is most often used when describing systems that are naturally periodic and not necessarily localized in space, such as crystals and solid state applications.

$$\text{Plane wave:} \quad \phi_{\boldsymbol{k}}(\boldsymbol{r}) \propto e^{i\boldsymbol{k}\boldsymbol{r}} \tag{29}$$

Here, $\boldsymbol{k}$ is a reciprocal lattice vector. We will explore the plane wave basis in more detail in section 4.3.

2.3.3 Classical Methods and Limitations

By introducing a finite basis set, the Schrödinger equation (20) becomes an eigenvalue equation between matrices and vectors. The most straightforward way of solving this type of equations is called Exact Diagonalization (ED) and consists of performing a basis transformation that diagonalizes the Hamiltonian. The energies then correspond to the diagonal entries of the Hamiltonian. The problem with this approach, however, is that the dimension of the Hilbert space and therefore the size of the matrices scales exponentially with the number of basis functions N. As we have seen before, this exponential scaling is a common theme in the classical treatment of quantum mechanics and the main reason why these approaches are only suitable for relatively small systems.

The Variational Principle

Due to this difficulty, the most common methods in quantum chemistry are based on the Rayleigh-Ritz variational principle. This principle states that the expectation value of the Hamiltonian in any state $|\psi\rangle$ is bounded from below by the lowest energy eigenvalue E_0, expressed as the inequality

$$\frac{\langle\psi| H |\psi\rangle}{\langle\psi|\psi\rangle} \geq E_0 \,. \tag{30}$$

This fact can then be used to find an approximate expression for the ground state $|\psi_0\rangle$ that minimizes the energy. Typically, one begins by expressing an initial Ansatz state $|\psi_{\boldsymbol{\theta}}\rangle$ through a set of parameters $\boldsymbol{\theta}$. By variationally optimizing the parameters one then obtains a state that minimize the left hand side of equation (30).

An example of a successful classical algorithm that utilizes the variational principle is the *Hartree-Fock theory*. Apart from the discretization error that comes with a finite basis set, this methods is based on the approximation that we can express the electronic wavefunction as a single Slater determinant. It essentially corresponds to the assumption that each electron is affected by the other electrons only through an effective potential and thus termed a *self-consistent field theory* approach. The variational parameters are

contained in the single electron basis functions and are optimized by solving the self-consistent Hartree-Fock equations. A detailed review can be found in [19].
Instead of focusing on the many-electron wavefunction, it is also possible to choose the electron density ρ as the main object of interest. *Density Functional theory* (DFT) treats the expectation values and wavefunction as functionals of the electron density. In particular, the Hohenberg-Kohn theorems [27] guarantee that the ground state energy is a unique functional of ρ and we may therefore apply the variational principle, now with respect to the electron density. As one of the main differences to Hartree-Fock, DFT methods include effects of electron correlations, which typically lead to more accurate results.

2.3.4 Fault Tolerant Quantum Algorithms

Despite being undoubtedly successful in several applications, these classical methods either suffer from the problems of exponential scaling or necessarily introduce strong approximations. As a consequence, even if there are more efficient classical algorithms, we are still limited to either small system size or low accuracy and any small change in these parameters will lead to a significantly higher computational effort. To lighten this pessimistic outlook, we now introduce ways in which quantum computers may be used to solve this problem and simulate molecular systems in an exponentially more efficient way. Before turning to variational quantum algorithms in section 3, we begin by outlining the *Quantum Phase Estimation* (QPE) algorithm [36] as one example of a fault tolerant method. That is, despite demonstrating a superior scaling as compared to classical methods, QPE still requires more quantum resources than are available in current and near-term technologies.

Quantum Fourier Transform
Like many other quantum algorithms, QPE is based on the Quantum Fourier Transform (QFT). Due to the exponential advantage over its classical counterpart, the QFT is probably one of the most important results in quantum computing and the key to methods such as Shor's algorithm, discrete logarithms, and the hidden subgroup problem [43]. In the following, we consider a quantum circuit of n qubits and denote the number of basis states by $N = 2^n$. Since each basis state $|j\rangle$ is given by a string of 0's and 1's, denoted by $|j_1 \dots j_n\rangle$, we may interpret it as a binary number and thus obtain the orthonormal basis $\{|0\rangle,\dots,|N-1\rangle\}$. The quantum Fourier transform on n qubits is then defined as the discrete Fourier transform on N numbers. More specifically, a basis state $|j\rangle$ undergoes the transformation

$$|j\rangle \to \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2\pi i j k/N} |k\rangle \ . \tag{31}$$

It is possible to rewrite this mapping in terms of the binary representation of j. In particular, the r.h.s. of equation (31) can be expressed as the following product state [43].

$$|j_1 \dots j_n\rangle \to \frac{1}{\sqrt{N}} \left(|0\rangle + e^{2\pi i 0.j_n} |1\rangle\right) \otimes \dots \otimes \left(|0\rangle + e^{2\pi i 0.j_1 j_2 \dots j_n} |1\rangle\right) \tag{32}$$

Crucially, this representation allows us to find a relatively simple quantum circuit that implements the quantum Fourier transform. To see this, we define the gate

$$R_k := \begin{pmatrix} 1 & 0 \\ 0 & e^{2\pi i/2^k} \end{pmatrix}. \tag{33}$$

After applying a Hadamard gate to the first qubit, we apply a sequence of $n-1$ R-gates to the same qubit, each one conditioned on one of the remaining qubits, leading to the first term in equation (32). We omit the normalization factors for clarity.

$$|j_1 \dots j_n\rangle \xrightarrow{H} \left(|0\rangle + e^{2\pi i 0.j_1} |1\rangle\right) |j_2 \dots j_n\rangle \xrightarrow{cR_2} \dots \xrightarrow{cR_n} \left(|0\rangle + e^{2\pi i 0.j_1 \dots j_n} |1\rangle\right) |j_2 \dots j_n\rangle \tag{34}$$

We continue analogously with the other qubits and generate the remaining terms in the product state. The resulting circuit is depicted in Fig. 2.

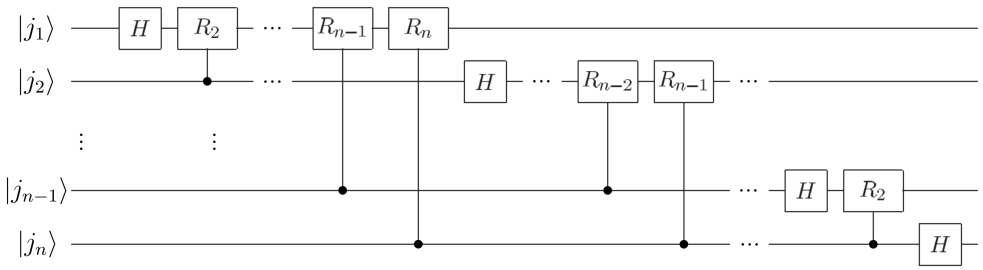


FIGURE 2: Circuit for the implementation of the Quantum Fourier Transform

From this construction it becomes clear that we can implement the QFT using $n(n+1)/2$ gates, implying a scaling of $\mathcal{O}(n^2)$. At first sight it might seem that we have not gained much compared to the classical Fourier transform, which scales quadratically with the number of points in the data set. The crucial point, however, is that the QFT does not transform n data points, but 2^n. It thus achieves an improvement from $\mathcal{O}(N^2)$ to $\mathcal{O}(\log^2 N)$. Unfortunately, since the transformed values are stored in the amplitudes of a quantum state, they are not directly accessible. The QFT can therefore not be used to speed up the Fourier transform of classical data and has to be used in a more subtle way, typically as a subroutine in a more complex algorithm.

Quantum Phase Estimation

One such algorithm that depends on the QFT is Quantum Phase Estimation. The aim of this algorithm is to efficiently estimate the eigenvalues of a unitary operator. More specifically, consider a unitary operator U acting on n qubits with eigenvector $|u\rangle$ and eigenvalue $e^{2\pi i\varphi}$, where the phase φ is assumed to be unknown and in the range $0 \leq \varphi \leq 1$. The QPE algorithm requires two registers: the first register stores t qubits initialized to $|0\rangle$, where the number t ultimately determines the decimal precision to which φ is computed. The second register consists of the n qubits that are necessary to store $|u\rangle$. We begin by applying a Hadamard gate to each qubit in the first register. Next, we assume that we have access to a black-box that efficiently implements controlled-U^{2^k} operations and apply them to the second register. More precisely, we iterate through the first register and for each qubit q we apply a controlled-$U^{2^{t-q}}$ gate conditioned on q and applied to the whole second register. We thus implement the following transformation:

$$|0\rangle^{\otimes t}|u\rangle \;\rightarrow\; \Big(|0\rangle+|1\rangle\Big)^{\otimes t}|u\rangle \;\rightarrow\; \Big(|0\rangle+e^{2\pi i 2^{t-1}\varphi}|1\rangle\Big)\ldots\Big(|0\rangle+e^{2\pi i 2^{0}\varphi}|1\rangle\Big)|u\rangle$$

$$= \Big(|0\rangle+e^{2\pi i 0.\varphi_t}|1\rangle\Big)\ldots\Big(|0\rangle+e^{2\pi i 0.\varphi_1\ldots\varphi_t}|1\rangle\Big)|u\rangle \tag{35}$$

Interestingly, we only act on the qubits in the first register indirectly by including them as control qubits. Yet, they are the only qubits that are affected by the above transformation. This process is referred to as *phase kick-back* and only possible because the second register was initialized in an eigenstate of U. The key step of the phase estimation algorithm is to recognize that the state of the first register in nothing else than the quantum Fourier transform of the state $|\varphi\rangle$. Accordingly, we can simply apply the inverse Fourier transform and read out the binary representation of φ through measurements in the computational basis. The circuit implementing the QPE algorithm is depicted in Fig. 3.

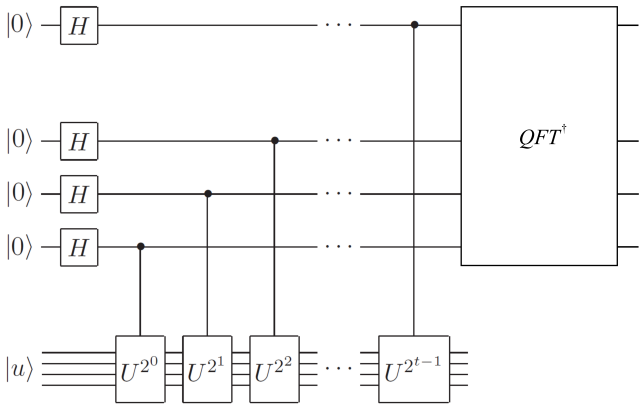


FIGURE 3: Circuit for the implementation of the Quantum Phase Estimation algorithm

There are two problems with the algorithm as we have described it so far. First, the phase φ can generally not be written as an exact binary representation of length t. A measurement in the computational basis can therefore only return an approximation and is not guaranteed to succeed. More precisely, in order to obtain φ accurate to r bits with a probability of success greater than $1 - \epsilon$, we have to choose

$$t \geq r + \left\lceil \log\left(2 + \frac{1}{2\epsilon}\right)\right\rceil . \tag{36}$$

A derivation of this result can be found in [43]. Second, we previously assumed that we can simply prepare the state $|u\rangle$ exactly. In general, however, we are only able to prepare an approximation $|\psi\rangle = \sum_k \lambda_k |u_k\rangle$. One can show that a computational basis measurement then results in the phase φ_k with probability $|\lambda_k|^2$.

As mentioned before, QPE is a key step in several important quantum algorithms. In particular, it can be used to compute the energies of the molecular Hamiltonian H and thus provide an efficient solution to the electronic structure problem. In order to transform this problem into the setting of QPE, we define the unitary operator $U = e^{-iH\tau/\hbar}$. This operator corresponds to the time evolution of a system described by H by time τ. Since H is given by a sum of non-commuting terms, a Trotter-Suzuki expansion [26] is typically used to split U into factors that can be easily implemented through elementary quantum gates. The energies E_k are then obtained by repeating the procedure several times for different values of τ and computing the variation of the phase φ_k.

$$E_k = -2\pi\hbar\frac{\Delta\varphi_k}{\Delta\tau} \tag{37}$$

Using a QPE based approach to quantum simulation, it would be possible to demonstrate quantum supremacy with a relatively small number of qubits [18]. However, the gate depth far exceeds the resources available to of any near-term quantum computer and thus QPE remains a method for the fault-tolerant future.

For exactly this reason, we now turn to the topic of variational quantum algorithms. VQAs typically combine classical and quantum resources in a way that enables relatively low gate depths and are thus candidates for useful algorithms in the NISQ era of quantum computing.

3 VQE in Second Quantization

In this chapter, we will give a detailed overview of the Variational Quantum Eigensolver as it is commonly used in quantum chemistry simulations. The purpose of this part is to provide a baseline that any proposed ideas in this thesis can then be compared to. We begin with an outline of the algorithm and then examine the different components, in particular regarding their influence on the overall scaling behaviour. A depiction of the structure of the algorithm can is presented in Fig. 4.

Outline of the VQE algorithm

$$\texttt{INPUT:} \begin{cases} \text{molecular configuration,} \\ \text{single-electron basis set} \end{cases} \qquad \texttt{OUTPUT:} \begin{cases} \text{ground state energy } E_0, \\ \text{ground state parameters } \theta_0 \end{cases}$$

`STEP 1:` Compute the overlap integrals h_{pq} and h_{pqrs} and discretize the Hamiltonian $H = \sum h_{pq} a_p^\dagger a_q + \sum h_{pqrs} a_p^\dagger a_r^\dagger a_q a_s$

`STEP 2:` Apply a fermion-to-qubit mapping to write the Hamiltonian in terms of Pauli strings that can be directly measured: $H = \sum_i c_i P_i$

`STEP 3:` Initialize the qubit register in a product state, typically resulting from a classical computation, e.g. Hartree-Fock (HF)

`STEP 4:` Apply a parameterized Ansatz circuit $U(\boldsymbol{\theta})$ to prepare the trial wavefunction $|\psi(\boldsymbol{\theta})\rangle$

`STEP 5:` Measure all Pauli strings P_i in H to obtain the energy expectation value $E(\boldsymbol{\theta}) = \langle H \rangle_{\psi(\boldsymbol{\theta})} = \sum_i c_i \langle P_i \rangle$

`STEP 6:` Adjust the parameters $\boldsymbol{\theta}$ using a classical optimization method (e.g. gradient descent)

`STEP 7:` Go back to `STEP 3` and repeat with updated parameters until convergence is achieved.

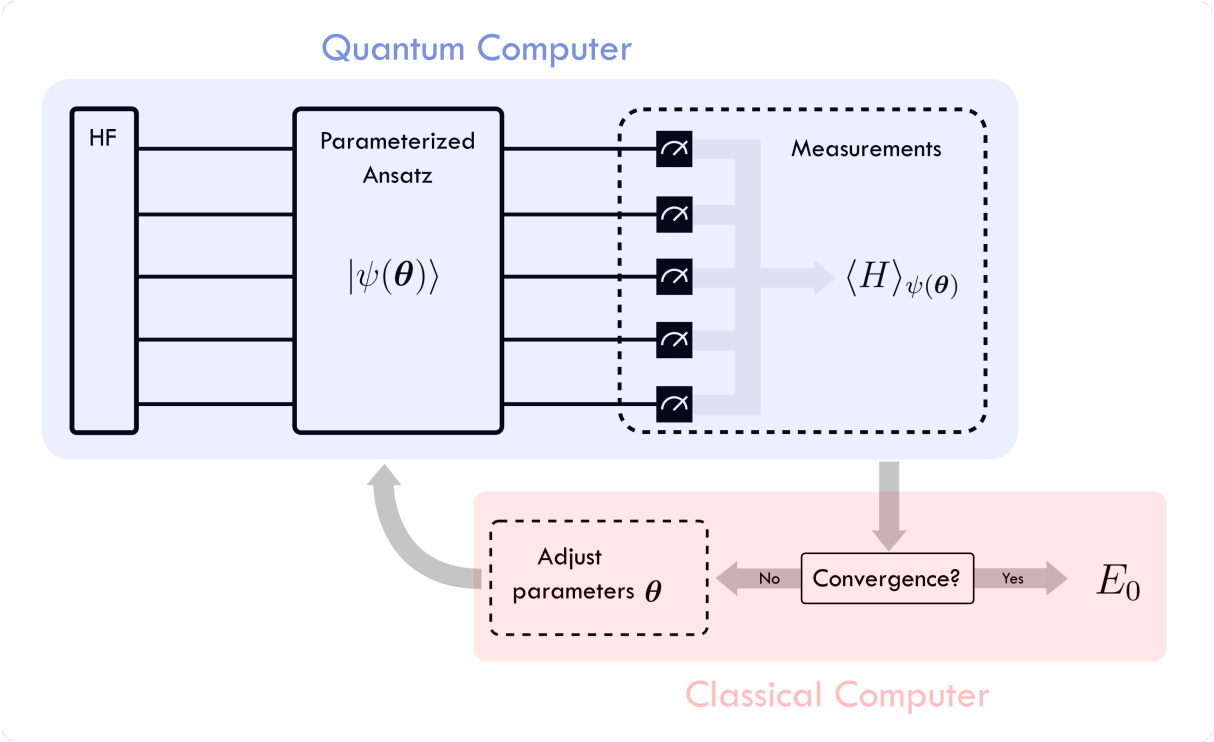


FIGURE 4: Outline of the Variational Quantum Eigensolver with highlighted parts that are performed on a quantum computer or classical computer

3.1 Hamiltonian Representation

Within quantum theory, electrons are treated as *indistinguishable* particles. In other words, exchanging any two electrons cannot have any measurable effect on the system and thus the joint probability distribution must remain constant.
Let $\psi(x_1, x_2)$ be the wavefunction describing two particles at positions x_1 and x_2. The condition of indistinguishability is then given by the requirement that

$$\left|\psi(x_1, x_2)\right|^2 = \left|\psi(x_2, x_1)\right|^2 . \tag{38}$$

We notice that this is still satisfied if the states differ by a complex phase $e^{i\varphi}$. It turns out, however, that in three-dimensional space there are only two possibilities: $e^{i\varphi} = +1$ for *bosons* and $e^{i\varphi} = -1$ for *fermions*. Electrons fall into the category of fermions and are thus described by wavefunctions that are antisymmetric under exchange of any two particles. Traditionally, enforcing the antisymmetry condition explicitly in the wavefunction is referred to as *First Quantization* and typically achieved by the use of Slater determinants as in equation (24).
The framework of *Second Quantization* differs from First Quantization in that antisym-

metry is enforced not through the wavefunction, but through the operators that generate the wavefunction. Whereas a Slater determinant essentially corresponds to enumerating each electron together with the orbital it occupies, we now work in the occupation number basis, where we keep a list of all orbitals and specify which one is occupied by an electron. In particular, we denote a state in the occupation number basis by $|\mu\rangle = |\mu_1, ..., \mu_N\rangle$, where $\mu_i = 1$ if the i-th orbital is occupied and $\mu_i = 0$ if the i-th orbital is unoccupied. Note that we have not made any explicit reference to which electron occupies which orbital. One the one hand this facilitates a very elegant way of describing states that are invariant under exchange of particles, one the other hand we now need some additional structure outside of the description of the state itself in order to distinguish between symmetric and antisymmetric particles. This additional structure is achieved by introducing creation operators $a_i^\dagger$ and annihilation operators a_i. In the fermionic case, these operators are defined in the following way:

$$a_i^\dagger |\mu_1, ..., \mu_N\rangle = \begin{cases} 0 & \text{if } \mu_i = 1 \\ p_i |\mu_1, ..., 1_i, ..., \mu_N\rangle & \text{if } \mu_i = 0 \end{cases} \tag{39}$$

$$a_i |\mu_1, ..., \mu_N\rangle = \begin{cases} 0 & \text{if } \mu_i = 0 \\ p_i |\mu_1, ..., 0_i, ..., \mu_N\rangle & \text{if } \mu_i = 1 \end{cases} \tag{40}$$

Here, p_i is the parity of the i-th orbital. In particular, $p_i = 1$ if the number of occupied orbitals up to but not including i is even and $p_i = -1$ if it is odd. One can show that this definition is equivalent to imposing the canonical fermionic anti-commutation relations

$$\left\{a_i, a_j\right\} = 0 \qquad \left\{a_i^\dagger, a_j^\dagger\right\} = 0 \qquad \left\{a_i^\dagger, a_j\right\} = \delta_{ij}\,, \tag{41}$$

where $\{A, B\} := AB + BA$. By defining the vacuum state $|\Omega\rangle = |0, ..., 0\rangle$ we can construct the state $|\mu\rangle$ simply by the action of a sequence of creation operators.

$$|\mu_1, ..., \mu_N\rangle = (a_1^\dagger)^{\mu_1} ... (a_N^\dagger)^{\mu_N} |\Omega\rangle \tag{42}$$

Formally, the states $|\mu\rangle$ are elements of the fermionic Fock space $\mathcal{H}_F = \mathcal{H}_0 \oplus \mathcal{H}_1 \oplus ... \oplus \mathcal{H}_N$, where $\mathcal{H}_k$ is the Hilbert space of all k-particle states. This structure is necessary to allow for superpositions of states with different particle numbers and treat them in the same state space.

We may now rewrite the discretized molecular Hamiltonian (25) using fermionic creation and annihilation operators. Intuitively, terms such as $\sum_i |\phi_p\rangle_i \langle\phi_q|_i$ represent excitations, i.e. the transfer of an electron from orbital q to orbital p. In Second Quantization, this action is realized by the operator $a_p^\dagger a_q$ and thus the Hamiltonian can be written as

$$H = \sum_{pq} h_{pq} a_p^\dagger a_q + \sum_{pqrs} h_{pqrs} a_p^\dagger a_r^\dagger a_q a_s \,. \tag{43}$$

3.2 Fermion-to-Qubit Encoding

A key step of the VQE algorithm is to compute the expectation value of the fermionic Hamiltonian (43) on a quantum computer. The immediate problem with this approach is that qubits as quantum 2-level systems are fundamentally different from fermions. In particular, from their definitions it becomes clear that the Fock space of N fermionic modes and the N-qubit Hilbert space are two different mathematical objects.
However, looking at equation (42) we notice that a particular configuration of m electrons in N fermionic modes is simply given by a string of 1s (for occupied orbitals) and 0s (for unoccupied orbitals). Conversely, a product state of N qubits is also described by a string of 1s (qubits in state $|1\rangle$) and 0s (qubits in state $|0\rangle$). It is therefore natural to introduce a mapping between the two spaces. As an example, consider a state of 3 fermionic modes, such that electrons occupy the first and third mode. The same binary string represents three qubits, such that the first and third qubit are in state $|1\rangle$ and the second qubit is in state $|0\rangle$.

$$|1,0,1\rangle_F \qquad \leftrightarrow \qquad |101\rangle_Q = |1\rangle_1 \otimes |0\rangle_2 \otimes |1\rangle_3 \tag{44}$$

Such a mapping between fermions and qubits was first introduced in 1928 by Jordan and Wigner [33] who showed that the space of N fermionic modes and the space of N qubits are in fact isomorphic. As a result, instead of working with fermionic states in the occupation number basis we are now able to map them to qubit states and make use of quantum computers to manipulate them and perform measurements.
In order to transform the Hamiltonian (43), we need to map the creation and annihilation operators $a_i^\dagger$ and a_j to qubit operators s_i^+ and s_j^- that have the same effect. For instance, since $a^\dagger |0\rangle_F = |1\rangle_F$ we want the qubit version of the operator to have the effect $s^+ |0\rangle_Q = |1\rangle_Q$. As a first guess, we might choose the qubit excitation operators

$$\sigma_i^+ = \mathbb{1} \otimes ... \otimes |1\rangle \langle 0|_i \otimes ... \otimes \mathbb{1} \qquad \text{and} \qquad \sigma_i^- = \mathbb{1} \otimes ... \otimes |0\rangle \langle 1|_i \otimes ... \otimes \mathbb{1} \tag{45}$$

Unfortunately, it not quite as simple as that. The key property that s_i^+ and s_j^- must satisfy are the anti-commutation relations (41). Although the definition above leads to the correct relations for the case $i = j$, it produces the wrong result in the general case. The reason for this behaviour is that single qubit excitation operators on different qubits always commute, but the non-local behaviour of fermions implies that creation operators on different particles do not commute.

$$\left\{\sigma_i^-, \sigma_i^-\right\} = 0 \qquad \left\{\sigma_i^+, \sigma_i^+\right\} = 0 \qquad \left\{\sigma_i^+, \sigma_i^-\right\} = \mathbb{1}\,, \tag{46}$$

$$\text{but e.g.} \quad \begin{cases} \left[\sigma_i^+, \sigma_j^+\right] = 0 \qquad \text{and} \qquad \left[a_i^\dagger, a_j^\dagger\right] = -2a_j^\dagger a_i^\dagger \neq 0 \\ \\ \left\{\sigma_i^+, \sigma_j^-\right\} = 2\sigma_i^+\sigma_j^- \neq 0 \quad \forall\, i \neq j \end{cases} \tag{47}$$

We therefore need to incorporate the non-local nature of fermionic operators into the definition of their qubit versions. In the following sections, we will introduce several ways of achieving the correct anti-commutation relations, beginning with the very prominent Jordan-Wigner transformation.
When deciding for a specific encoding, there are three main characteristics that need to be considered. First, the number of qubits needed to store the fermionic wavefunction may differ between the encodings and is naturally kept as small as possible. Second, the Pauli weight is defined as the maximum number of qubits affected by a single Pauli string. A lower Pauli weight typically leads to Ansatz circuits with lower depth and smaller read-out error. Third, the number of Pauli strings in the transformed Hamiltonian is very important, as it determines the number of measurements needed to compute the energy in each VQE iteration.

3.2.1 Jordan-Wigner Encoding

In order to define the different encodings in a consistent way, we introduce a vector notation for both the fermionic state $|\mu\rangle$ in the occupation number basis and the qubit state $|q\rangle$.

$$\vec{\mu} := \begin{pmatrix} \mu_1 \\ \vdots \\ \mu_N \end{pmatrix} \qquad \vec{q} := \begin{pmatrix} q_1 \\ \vdots \\ q_N \end{pmatrix} \tag{48}$$

Any (linear) transformation between the two spaces can then be represented as a matrix Λ. Hence, we may formally write a fermion-to-qubit mapping as a linear equation between the vectors $\vec{\mu}$ and $\vec{q}$.

$$\vec{q} = \Lambda\vec{\mu} \mod 2\,. \tag{49}$$

As already mentioned above, the Jordan-Wigner encoding is simply given by a one-to-one correspondence between the occupancy of the i-th orbital and the state of the i-th qubit. Using the vector notation, the Jordan-Wigner transformation Λ_{JW} then coincides with the $N \times N$ identity matrix.

$$\Lambda_{\mathrm{JW}} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & & \\ \vdots & & \ddots & \\ 0 & & & 1 \end{pmatrix} \tag{50}$$

Next, we need to define qubit versions of the creation and annihilation operators that satisfy the anti-commutation relations (41). As we have seen before, simply choosing the excitation operators $\sigma_i^+ = |1\rangle\langle 0|_i$ and $\sigma_i^- = |0\rangle\langle 1|_i$ does not account for the non-local properties of fermions. In order to include the parity p_i from definitions (39) and (40) we observe that the operator $Z_1 \otimes ... \otimes Z_{i-1}$ has eigenvalues that are identical to the parity. By adding this Z-string to the qubit excitation operators we can now define the Jordan-Wigner mapping

$$a_i^\dagger \mapsto s_i^+ := \left(\prod_{k=1}^{i-1} Z_k\right)\sigma_i^+ \qquad \text{and} \qquad a_i \mapsto s_i^- := \left(\prod_{k=1}^{i-1} Z_k\right)\sigma_i^-\,. \tag{51}$$

It is straightforward to confirm that this reproduces the correct anti-commutation relations (see Appendix A.1). The definition (51) implies that $a_i^\dagger a_i = \sigma_i^+\sigma_i^- = |0\rangle\langle 0|_i = (\mathbb{1} + Z_i)/2$, thus enabling us to write the inverse transformation in the following way:

$$s_i^+ \mapsto \prod_{k=1}^{i-1}\left(\mathbb{1} - 2a_k^+ a_k\right)a_i^\dagger \qquad s_i^- \mapsto \prod_{k=1}^{i-1}\left(\mathbb{1} - 2a_k^+ a_k\right)a_i \qquad Z_i \mapsto \mathbb{1} - 2a_i^\dagger a_i \tag{52}$$

Using equation (51) we may now rewrite the Hamiltonian (43) in the qubit space in terms of the operators s_i^+ and s_i^-. In this form, H can easily expanded as a sum of Pauli strings that can be measured directly on a quantum computer. The expectation value $\langle H \rangle$ can be computed by linearly combining the Pauli string expectation values.

$$H = \sum_{pq} h_{pq} s_p^+ s_q^- + \sum_{pqrs} h_{pqrs} s_p^+ s_r^+ s_q^- s_s^- =: \sum_l \tilde{h}_l P_l \tag{53}$$

$$\Rightarrow \langle H \rangle = \sum_l \tilde{h}_l \langle P_l \rangle \tag{54}$$

A direct consequence of including a Z-string in the transformation is that the Pauli-weight of the Jordan-Wigner encoding scales linearly with the number of orbitals as $\mathcal{O}(N)$.
The operators $\sigma_i^\pm$ can be expressed using the two Pauli matrices X and Y such that $\sigma_i^\pm = (X \mp iY)/2$. As a consequence, each fermionic operator is mapped to a sum of two Pauli strings and thus the total number of Pauli strings in the transformed Hamiltonian is proportional to the number of fermionic operators. From equation (43) it becomes clear that the molecular Hamiltonian consists of $\mathcal{O}(N^4)$ terms, resulting in $\mathcal{O}(N^4)$ Pauli strings.

3.2.2 Parity Encoding

Rather than storing the occupancy of orbitals in the qubits directly via the Jordan-Wigner transformation, it is possible to use a more indirect approach and store the orbital parity instead. Using the definition from section 3.1, the parity p_i of orbital i can be written as

$$p_i = \sum_{k=1}^{i-1} \mu_k \mod 2\,. \tag{55}$$

In this encoding, we set the qubit q_i equal to the parity p_i. In vector notation, the transformation may then be represented as the matrix

$$\Lambda_\mathrm{P} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 1 & 1 & & \\ \vdots & & \ddots & \\ 1 & 1 & \cdots & 1 \end{pmatrix}, \tag{56}$$

where all entries below the diagonal are equal to 1. The information about orbital occupancy can be easily reconstructed from a given qubit state $\vec{q}$. Namely, the i-th orbital

was occupied if $q_i \neq q_{i+1}$, implying that the parity changes with this index. Conversely, the orbital was unoccupied if the parity stays the same.
A consequence of using the parity encoding is that we do not need the Z-string in the transformed qubit excitation operators. The purpose of this string was to incorporate a phase factor corresponding to the parity, such that the correct anti-commutation relations are satisfied. In the parity encoding, however, the parity p_{i-1} needed for the operators s_i^+ and s_i^- is already stored in the qubit q_{i-1}. It might therefore seem sufficient to include a single operator Z_{i-1} in addition to the excitation operators σ_i^+ and σ_i^-. However, since the effect of these operators is to add or remove an electron, the parity for all indices $k > i$ changes, and we need to update the state of all qubits with those indices. This can be achieved by adding an X-string to every qubits with index $k > i$.
Furthermore, we now need to distinguish more carefully between the operators σ_i^+ and σ_i^-. In the JW-encoding, the creation operator $a_i^\dagger$ always implies the use of the excitation operator, σ_i^+ and similarly a implies the use of σ_i^-. In the parity encoding, however, this is only the case if the parity $p_{i-1} = 0$. If $p_{i-1} = 1$, the correspondence reverses and now $a_i^\dagger$ implies the use of σ_i^-. We can include this condition by using the projections operators

$$|0\rangle\langle 0|_{i-1} = \frac{1}{2}\left(\mathbb{1} + Z_{i-1}\right) \qquad \text{and} \qquad |1\rangle\langle 1|_{i-1} = \frac{1}{2}\left(\mathbb{1} - Z_{i-1}\right) \tag{57}$$

Finally, the resulting transformation rule for the creation operator can be simplified by using the definition $\sigma^\pm = (X \mp iY)/2$.

$$\begin{aligned} a_i^\dagger \mapsto s_i^+ :&= Z_{i-1}\left(|0\rangle\langle 0|_{i-1} \otimes \sigma_i^+ + |1\rangle\langle 1|_{i-1} \otimes \sigma_i^-\right)\left(\prod_{k=i+1}^{N} X_k\right) \\ &= \left(|0\rangle\langle 0|_{i-1} \otimes \sigma_i^+ - |1\rangle\langle 1|_{i-1} \otimes \sigma_i^-\right)\left(\prod_{k=i+1}^{N} X_k\right) \\ &= \frac{1}{2}\left(Z_{i-1} \otimes X_i - i\mathbb{1}_{i-1} \otimes Y_i\right)\left(\prod_{k=i+1}^{N} X_k\right) \end{aligned} \tag{58}$$

$$\Rightarrow a_i \mapsto s_i^- := \frac{1}{2}\left(Z_{i-1} \otimes X_i + i\mathbb{1}_{i-1} \otimes Y_i\right)\left(\prod_{k=i+1}^{N} X_k\right) \tag{59}$$

Due to the inclusion of the X-string, the Pauli weight exhibits the same scaling $\mathcal{O}(N)$ as the Jordan-Wigner transformation. Similarly, the number of Pauli strings is unchanged and scales as $\mathcal{O}(N^4)$.

3.2.3 Bravyi-Kitaev Encoding

A natural question is whether there are any transformations with lower Pauli weight than the Jordan-Wigner and Parity encoding. In 2002, Bravyi and Kitaev [17] first showed that this is in fact possible, and the resulting encoding achieves a reduction in the Pauli weight from $\mathcal{O}(N)$ to $\mathcal{O}(\log_2 N)$.
The Bravyi-Kitaev encoding stores a combination of parity and occupation number and thus integrates aspects from both the JW and parity encoding. The transformation matrix is defined recursively, starting with the smallest element $\Lambda_0 = 1$. Each successive step is then constructed using the following rule.

$$\Lambda_{2^k} = \left(\begin{array}{c|c} \Lambda_{2^{k-1}} & 0 \\ \hline 0 & \\ \leftarrow 1 \rightarrow & \Lambda_{2^{k-1}} \end{array}\right) \qquad \Lambda_0 = (1) \tag{60}$$

Note that the size of each matrix doubles with each iteration, justifying the index 2^k. In order to construct the transformation matrix for an arbitrary N, one simply generates $\Lambda_{2^{k_0}}$ such that $2^{k_0} \geq N$ and discards all rows and columns with indices larger than N.
Consider the first 4 iterations of matrices. It becomes clear that different qubits now play different roles in the encoding. For instance, qubits with even indices still store the occupancy directly, indicated by matrix rows that are empty except for a 1 on the diagonal. The remaining qubits store the parity of orbitals within smaller groups of varying size.

$$\Lambda_0 = (1) \quad \Lambda_2 = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \quad \Lambda_4 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix} \quad \Lambda_8 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Writing down expressions for the transformed operators is not as straightforward as in the previous cases and requires some preparation. In order to simplify the notation, we write $A_S = \prod_{i \in S} A_i$ for an operator A acting on a set of qubit indices S. Furthermore, it is convenient to define the following sets of qubits:

- The **Update set** $U(i)$: this set contains all qubits that depend on the occupancy of orbital i. The j-th qubit is an element of $U(i)$ if a change in the orbital i changes the value q_j. In the parity encoding, the Update set is the set of qubits that the X-string acts on.

- The **Parity set** $P(i)$: the qubits in $P(i)$ encode the parity of orbitals up to but not including i. In the Bravyi-Kitaev encoding there is no longer one single qubit that stores the full parity each orbital. Rather, the parity must be constructed using several different qubits, each encoding the parity of a subset of orbitals. In other words, $\bigoplus_{j\in P(i)} q_j = \bigoplus_{j=1}^{i-1} \mu_j = p_j$. In the parity encoding, $P(i) = \{i-1\}$ and in the JW-encoding, $P(i) = \{1, ..., i-1\}$.

- The **Flip set** $F(i)$: the qubits in $F(i)$ determine whether the value of qubit q_i is equal to or different from the occupancy of orbital i. In the case of the parity encoding, whether a creation operator a^+ implies the use of σ^+ or σ^- depends on the qubit q_{i-1} storing the parity p_{i-1}. In general, and specifically in the Bravyi-Kitaev encoding, qubits in the Flip set only need to store the parity within each sub-block, implying that $F(i) \subset P(i)$.

- The **Remainder set** $R(i)$: this set is simply defined as the set of qubits that are in the Parity set but not in the Flip set, i.e. $R(i) = P(i) \backslash F(i)$.

Note that these definitions do not depend on any specific encoding and can thus be used to derive a completely general transformation rule for creation and annihilation operators. In a straightforward way, we apply X operators to each qubit in the Update set, use the flip set to decide whether to apply σ^+ or σ^-, and apply Z operators to each qubit in the parity set. The qubits in the Flip set are now exactly those that are affected by both a Z operator and a projection operator $|0\rangle\langle 0|$ or $|1\rangle\langle 1|$.

$$
\begin{aligned}
a_i^\dagger \mapsto s_i^+ :&= Z_{P(i)} \otimes \Big(|0\rangle\langle 0|_{F(i)} \otimes \sigma_i^+ - |1\rangle\langle 1|_{P(i)} \otimes \sigma_i^- \Big) \otimes X_{U(i)} \\
&= Z_{R(i)} \otimes \Big(|0\rangle\langle 0|_{F(i)} \otimes \sigma_i^+ + |1\rangle\langle 1|_{F(i)} \otimes \sigma_i^- \Big) \otimes X_{U(i)} \\
&= \frac{1}{2}\Big(Z_{R(i)} \otimes (\mathbb{1} + Z_{F(i)}) \otimes \sigma_i^+ \otimes X_{U(i)} + Z_{R(i)} \otimes (\mathbb{1} - Z_{F(i)}) \otimes \sigma_i^+ \otimes X_{U(i)} \Big) \\
&= \frac{1}{2}\Big(Z_{P(i)} \otimes \underbrace{\big(\sigma_i^+ + \sigma_i^-\big)}_{X_i} \otimes X_{U(i)} + \underbrace{Z_{P(i)} \otimes Z_{F(i)}}_{Z_{R(i)}} \otimes \underbrace{\big(\sigma_i^+ - \sigma_i^-\big)}_{-iY_i} \otimes X_{U(i)} \Big) \\
&= \frac{1}{2}\Big(Z_{P(i)} \otimes X_i \otimes X_{U(i)} - i Z_{R(i)} \otimes Y_i \otimes X_{U(i)} \Big)
\end{aligned}
$$

To summarize, the transformation rule for any arbitrary encoding is given by

$$\begin{aligned} s_i^+ &= \frac{1}{2}\Big(Z_{P(i)} \otimes X_i \otimes X_{U(i)} - iZ_{R(i)} \otimes Y_i \otimes X_{U(i)}\Big) \\ s_i^- &= \frac{1}{2}\Big(Z_{P(i)} \otimes X_i \otimes X_{U(i)} + iZ_{R(i)} \otimes Y_i \otimes X_{U(i)}\Big) \end{aligned} \tag{61}$$

Unfortunately, the recursive nature of the definition of the Bravyi-Kitaev transformation prevents us from achieving a more specific representation for this encoding. However, one now simply has to generate the correct sets $P(i)$, $U(i)$, and $R(i)$ for each qubit and plug them into equation (61).

3.2.4 Optimal Encodings

Even though the Jordan-Wigner, Parity, and Bravyi-Kitaev encodings are very well known and widely used, there is a variety of other encodings that have advantages in certain situations. As an example, we briefly mention optimal encodings based on ternary trees [32]. By expanding the Hamiltonian in terms of Majorana operators and mapping the qubits to vertices of a ternary tree, an asymptotically optimal Pauli weight of $\mathcal{O}(\log_3(2n+1))$ is achieved. This reduced Pauli weight typically does not substantially affect the read-out error of measurements [57], as the efficient measurement grouping strategies require a measurement of the entire register. The main advantage of the reduced Pauli weight is that it affects the gate depth of several Ansatz circuits, such as the Unitary Coupled Cluster Ansatz [38] and the ADAPT-VQE Ansatz [25].

3.3 Ansatz Choice

Now that we have mapped the molecular Hamiltonian to qubit space and generated a list of Pauli strings that need to be measured, we turn to the second important step in the VQE algorithm: the preparation of the Ansatz state. In practice, we have the freedom to choose the number, location, and type of quantum gates in the Ansatz circuit.
In order to choose a suitable Ansatz there are two important criteria that have to be evaluated. On the one hand, the *expressibility* of the Ansatz describes its ability to accurately approximate the true ground state of the system. On the other hand, the *trainability* of the Ansatz describes our ability to efficiently optimize its parameters with respect to the expectation value of the Hamiltonian as cost function. Mathematically, an Ansatz is considered trainable if the gradient in the optimization procedure vanishes at least polynomially with the number of qubits and the circuit depth. Of course, expressibility and trainability are somewhat conflicting requirements. A more expressible Ansatz can be achieved by increasing the number of gates and parameters in the circuit, decreasing its trainability as a result. Finding the optimal Ansatz therefore requires us to accept a trade-off between both criteria.
A common problem that VQE Ansätze suffer from is the *barren-plateau* problem [57]. By definition, it occurs if the energy gradient in the optimization procedure vanishes exponentially with the circuit depth and number of qubits, thus making the minimization intractable. As real quantum computers are subject to noise, a barren plateau makes it increasingly difficult to distinguish between positive and negative gradients. In real applications, it is therefore desirable to maintain trainability and accept the cost of reduced expressibility.
The first step in the state preparation is to initialize the qubits in a reference state. This state is typically obtained from a previous classical calculation and serves as an initial guess for the ground state. In particular, the reference state should introduce the correct symmetries of the problem, i.e. particle number, total spin, and the z-component of the spin. In practice, this step is implemented by simple bit-flip operations on an initial $|0\rangle^{\otimes N}$ state.
VQE Ansätze can be divided into several categories. Mainly, one distinguishes between *fixed structure* (or static) Ansätze, which have a circuit architecture that is predetermined and independent of the problem at hand, and *adaptive structure* (or non-static) Ansätze, which are build iteratively throughout the algorithm and adapt to the problem at hand.

3.3.1 Fixed Structure Ansätze

Fixed structure Ansätze are given by a set of gates whose locations are determined at the beginning of the algorithm and remain unchanged throughout the optimization process. Of course, the action of each gate still depends on a parameter θ which is adjusted in order

to minimize the energy. Within this category, Ansätze are further divided into *problem agnostic* and *problem tailored* ones. As the name suggests, problem agnostic Ansätze do not attempt to use any information that is specific to the molecular system under consideration. Typically, Ansätze of this kind focus on a hardware-efficient implementation, only using gates that are natural to the specific quantum computing hardware used in the experiment. Problem tailored Ansätze, on the other hand, aim to incorporate some information about the molecular system, such as symmetries or spin properties. In the following, we will briefly outline two choices within these categories.

Hardware-Efficient Ansatz

The Hardware-Efficient Ansatz (HEA) is an example of a fixed structure, problem agnostic Ansatz. The motivation behind this method is to use parameterized gates that are tailored to the device it is implemented on. The advantage of this approach is that, by assumption, each gate can be easily implemented and thus the available gate depth is used efficiently. On the other hand, as it is a problem agnostic method, HEA requires a high degree of expressibility, resulting in a large number of gates and parameters. In fact, in the worst case scenario, the number of parameters can grow exponentially with the size of the system. As a result, typical HEA approaches suffer extensively from the barren-plateau problem and quickly become intractable.

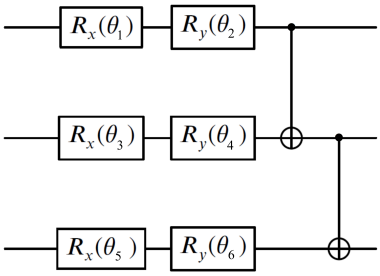


FIGURE 5: An example of a circuit implementing one block of a Hardware-Efficient Ansatz. The specific choice of rotation and entangling gates depends on the device they are implemented on.

Although there are many variations of HEA, they typically follow the same basic structure. The circuit is constructed by repeatedly applying blocks of parameterized single-qubit rotation gates and ladders of entangling gates. A possible example of this structure is shown in Fig. 5. The specific choice of rotation and entangling gate depends on the quantum computer architecture. Due to its effective gate implementation, the HEA Ansatz has been used in several small-scale experiments [34]. However, due to the barren-plateau problem it is not a suitable candidate for mid-scale or large-scale applications.

Unitary Coupled Cluster Ansatz

The Unitary Coupled Cluster (UCC) Ansatz is the most prominent example of a fixed structure, problem tailored Ansatz. It featured in the original publication of VQE by Peruzzo et al. [46] and has been a prominent Ansatz in the VQE literature since. By including information about the molecular system at hand, we are able to exclude regions of the Hilbert space in which we do not expect the true ground state to be contained. Even though this method technically reduces the expressibility of the Ansatz, it ideally does not substantially reduce the precision of the simulation result.

The UCC Ansatz originates from Coupled Cluster theory, which is a post Hartree-Fock method with the aim of including some of the electron correlation energy that is lost in Hartree-Fock theory. Within this method, an initial wavefunction is evolved under the action of parameterized excitation operators. The resulting CC Ansatz function is given by

$$\left|\psi^{\mathrm{CC}}\right\rangle = e^{\hat{T}} \left|\psi^{HF}\right\rangle . \tag{62}$$

The cluster operator $\hat{T}$ is expanded up to the desired degree of excitations.

$$T = T_1 + T_2 + T_3 + ... \tag{63}$$

Typically, we only keep the single and double excitation terms, given by

$$T_1 = \sum_{ia} \hat{t}_i^a = \sum_{ia} t_i^a a_a^\dagger a_i \qquad T_2 = \sum_{\substack{i<j \\ a<b}} \hat{t}_{ij}^{ab} = \sum_{\substack{i<j \\ a<b}} t_{ij}^{ab} a_a^\dagger a_b^\dagger a_i a_j \, . \tag{64}$$

Here, we use the indices i, j for occupied orbitals and a, b for unoccupied, or virtual, orbitals. Since the operator $e^{\hat{T}}$ is not unitary, we cannot simply implement it using quantum gates on a quantum computer. Unitary Coupled Cluster (UCC) theory was developed to solve this problem and is based on the fact that from any linear operator $\hat{T}$, we can construct an anti-hermitian operator $\hat{T} - \hat{T}^\dagger$. Since the exponential of any anti-hermitian operator is unitary, we may use the operator $e^{\hat{T}-\hat{T}^\dagger}$ as a parameterized unitary that can be implemented on a quantum circuit:

$$\left|\psi^{\mathrm{UCC}}\right\rangle = e^{\hat{T}-\hat{T}^\dagger} \left|\psi^{HF}\right\rangle . \tag{65}$$

Again, it is common practice to only keep the single and double excitation terms. In that case, the Ansatz is referred to as UCCSD (Unitary Coupled Cluster Singles Doubles).

Using the notation introduced in equation (64), we define the anti-hermitian excitation operators $\hat{\tau}_i^a$ and $\hat{\tau}_{ij}^{ab}$ and thus rewrite the unitary operator $e^{\hat{T}-\hat{T}^\dagger}$.

$$\hat{\tau}_i^a := \hat{t}_i^a - \hat{t}_a^i \quad \text{and} \quad \hat{\tau}_{ij}^{ab} := \hat{t}_{ij}^{ab} - \hat{t}_{ab}^{ij} \tag{66}$$

$$\Rightarrow e^{\hat{T}-\hat{T}^\dagger} = e^{\sum_{ia}\hat{\tau}_i^a + \sum_{ijab}\hat{\tau}_{ij}^{ab}} \tag{67}$$

In order to use this Ansatz in the context of VQE, we need to implement the unitary operator (67) on a quantum circuit. Since any realistic model of quantum computing requires each gate to act on only a few qubits at a time, the operator $e^{\hat{T}-\hat{T}^\dagger}$ must be broken down into a sequence of local gates. Unfortunately, as the single and double excitation operators do not commute, we cannot simply write the exponential (67) as a product of exponentials without introducing an approximation. Rather, we need to used the Trotter expansion for the exponential of operators A and B,

$$e^{A+B} = \lim_{n\to\infty}\left(e^{A/n}e^{B/n}\right)^n . \tag{68}$$

In practice, this expansion must be truncated at a finite value of n, thus introducing an approximation with respect to the original UCCSD Ansatz. However, it can been shown that even a single Trotter step $n = 1$ only results in a small deviation from the exact results [44]. In this case, the trotterized UCCSD (tUCCSD) Ansatz can be written as a product of exponentials of individual excitation operators:

$$\left|\psi^{\text{tUCCSD}}\right\rangle = \prod_{ia} e^{\hat{\tau}_i^a} \prod_{\substack{ij \\ ab}} e^{\hat{\tau}_{ij}^{ab}} \left|\psi^{\text{HF}}\right\rangle . \tag{69}$$

Under a fermion-to-qubit mapping, each excitation operator turns into a constant number of Pauli strings P_k. Each exponential $e^{\hat{\tau}}$ can thus again be approximated by a product of exponentials of these Pauli strings. It can be shown that the required circuit depth of this Ansatz scales polynomially in the system size. More specifically, each Trotter step requires a depth of $\mathcal{O}((N-m)^2 m)$, where m is the number of electrons and N the number of single electron orbitals [38]. There are several variations of the UCC-Ansatz, an overview of which can be found in [57].

So far we have treated the trotterized version of the UCCSD Ansatz as an approximation to the general UCCSD Ansatz. It is important to note, however, that technically this is only true if the parameters are optimized before the Trotterization. If we implement the unitary as given in equation (69) and then optimize its parameters, this Ansatz can be considered as a unique Ansatz that is different from the underlying UCC theory. In

fact, it can be shown that the Full Configuration Interaction (FCI) ground state can be achieved by expressed using an arbitrary long products of operators of the form (69) [57]. The reason for this is that n-body interactions can be expressed as a product of one- and two-body interactions, i.e.,

$$\left|\psi^{\mathrm{FCI}}\right\rangle = \prod_{k=0}^{\infty} \left(\prod_{ia} e^{\hat{\tau}_i^a(k)} \prod_{\substack{ij \\ ab}} e^{\hat{\tau}_{ij}^{ab}(k)} \right) \left|\psi^{\mathrm{HF}}\right\rangle \ . \tag{70}$$

The operators $\hat{\tau}_i^a(k)$ and $\hat{\tau}_{ij}^{ab}(k)$ are identical to the ones introduced in equation (66), with the exception that the parameters now depend on the index k. As a result, the expression (70) is no longer a Trotter expansion of any specific UCC Ansatz.

3.3.2 Adaptive Structure Ansätze

In the previous section, we have seen that the true ground state can be reached by the action of an infinite product of exponentials of single and double excitation operators, each of which can easily be implemented in terms of quantum gates. By truncating this infinite product at some finite value of k, we may therefore expect to achieve a better approximation of the true ground state than we could achieve using the UCC Ansatz.
Furthermore, it has been shown that *i*) the operator ordering in (69) affects the approximation error of the Ansatz [24], and *ii*) different operators contribute differently to the overall expressibility of the Ansatz [59]. It is therefore reasonable to expect that we can increase the efficiency of the Ansatz by ordering the operators differently and discarding those that contribute only a small amount to the overall expressibility. This is the motivation behind adaptive structure Ansätze, most prominently represented by ADAPT-VQE [25] and qubit-ADAPT-VQE [55].

ADAPT-VQE

Adaptive structure Ansätze differ from fixed structure Ansätze in that the number of gates in the Ansatz is not fixed. Rather, the Ansatz is built iteratively throughout the optimization process and new operators are added based on their ability to reduce the energy expectation value. In particular, we define an operator pool $\{\hat{A}_i\}$ from which one operator $\hat{A}_i$ is chosen in each iteration. Starting from the initial HF state $\left|\psi^{\mathrm{HF}}\right\rangle$, the ADAPT-VQE Ansatz is extended by multiplying from the left the operator $e^{\theta_i \hat{A}_i}$, where θ_i is a variational parameter.

$$\begin{aligned}
\left|\psi^{(0)}\right\rangle &= \left|\psi^{\mathrm{HF}}\right\rangle \\
\left|\psi^{(1)}\right\rangle &= e^{\theta_1 \hat{A}_1} \left|\psi^{\mathrm{HF}}\right\rangle \\
\left|\psi^{(2)}\right\rangle &= e^{\theta_2 \hat{A}_2} e^{\theta_1 \hat{A}_1} \left|\psi^{\mathrm{HF}}\right\rangle \\
&\vdots \\
\left|\psi^{(k)}\right\rangle &= e^{\theta_k \hat{A}_k} \ldots e^{\theta_1 \hat{A}_1} \left|\psi^{\mathrm{HF}}\right\rangle
\end{aligned} \tag{71}$$

Evidently, choosing a suitable operator pool and cutoff value k are crucial for the performance of the Ansatz. In the original publication [25], the operators $\hat{A}$ are chosen to be exactly the single- and double-excitation operators $\hat{\tau}_i^a$ and $\hat{\tau}_{ij}^{ab}$ used in UCC.
In order to decide which operator to choose in each iteration, it is proposed in [25] to pick the operator which results in the largest energy gradient with respect to the parameter associated with it. In particular, the energy gradient can be calculated as

$$\frac{\partial E^{(k)}}{\partial \theta_k} = \left\langle \psi^{(k)} \right| \left[H, \hat{A}_k \right] \left| \psi^{(k)} \right\rangle \ . \tag{72}$$

Since H and $\hat{A}_k$ can both be expanded as a sum of Pauli strings, evaluating the r.h.s of equation (72) amounts to measuring separate expectation values of Pauli strings. The full algorithm then consists of the following steps:

`STEP 1:` Compute the electron integrals h_{pq} and h_{pqrs} and transform the Hamiltonian under a fermion-to-qubit mapping

`STEP 2:` Define an operator pool $\{\hat{A}_i\}$ consisting of operators that can be used to extend the Ansatz. Typically, these are the generalized excitation operators $\hat{\tau}_i^a$ and $\hat{\tau}_{ij}^{ab}$. Then initialize the Ansatz $U_A^{(0)}$ to the identity operator

`STEP 3:` Initialize the qubit register in a reference state (e.g. HF) and prepare a trial state using the current Ansatz $U_A^{(k)}$

`STEP 4:` Compute the gradient $\frac{\partial E^{(k)}}{\partial \theta_k}$ for each operator in the pool for the current trial state $\left|\psi^{(k)}\right\rangle$, by measuring the Pauli string expectation values in the commutator $[H, \hat{A}_k]$

`STEP 5:` Choose the operator $\hat{A}_{k+1}$ with the largest gradient and update the current Ansatz by multiplication from the left using a new variational parameter θ_{k+1}, i.e. $U_A^{(k+1)} = e^{\theta_{k+1}\hat{A}_{k+q}} U_A^{(k)}$.

`STEP 6:` Perform a standard VQE-routine to optimize all parameters $\theta_1, ..., \theta_{k+1}$

`STEP 7:` Go back to `STEP 3` and repeat until the largest gradient is below a threshold ε

A good overview of the algorithm can be found in Fig. 1 in [25].
The structure of ADAPT-VQE clearly implies that the number of measurements is significantly higher than in traditional VQE. This is because it requires both a gradient measurement for every operator in the pool and a full VQE optimization of the current Ansatz in every iteration of the algorithm. In order to roughly estimate the number of measurements, we observe that there are $\mathcal{O}(N^4)$ elements in the operator pool, each of which can be expressed as a constant sum of Pauli strings. Since the Hamiltonian consists of $\mathcal{O}(N^4)$ terms as well, the commutator $[H, \hat{A}_i]$ turns into a sum of $\mathcal{O}(N^8)$ Pauli strings. Performing `STEP 4` of the algorithm thus requires determining $\mathcal{O}(N^8)$ Pauli string expectation values. As in the regular version of VQE, however, this number can be reduced by efficient measurement and grouping strategies [57].
Numerical simulation show [25], however, that ADAPT-VQE outperforms VQE based on the UCCSD Ansatz in terms of circuit depth and chemical accuracy. In particular, the same level of accuracy can be achieved using a smaller number of parameters, thus mitigating the barren-plateau problem.

qubit-ADAPT-VQE

Even though ADAPT-VQE achieves a lower overall gate depth compared to fixed structure Ansätze, each operator in the Ansatz still requires a significant number of elementary gates to implement. In fact, it is shown in [55] that in the limit of large N, the number of CNOT gates for each operator $e^{\theta\hat{\tau}_{ij}^{ab}}$ can be estimated to be approximately $64N$. The recently proposed qubit-ADAPT-VQE algorithm [55] is a variation of the ADAPT-VQE algorithm which addresses this problem and aims to reduce both the gate depth per operator and the number of operators in the pool.
The difference between ADAPT-VQE and qubit-ADAPT-VQE lies predominantly in the definition of the operator pool. Under a fermion-to-qubit mapping, each excitation operator $\hat{\tau}$ is mapped to a maximum of 8 Pauli strings P_i. In order to implement the operator $e^{\theta\hat{\tau}}$, it is therefore typically approximated using the trotterized product $\prod_i e^{i\theta P_i}$. We now define the operator pool in qubit-ADAPT-VQE as the set of exactly those Pauli strings P_i

that appear in the expansion of the excitation operators $\hat{\tau}$. This change has two important consequences. First, since the Ansatz is now extended by exponentials $e^{i\theta P_i}$ involving only a single Pauli string, this step is significantly cheaper to implement with respect to the circuit depth. And second, as argued in [55], if we assume the Hamiltonian to be symmetric with respect to time reversal, we can discard all Pauli strings that contain an even number of Y's as they do not affect the energy.

On the other hand, by choosing individual Pauli strings as pool operators we dispense with the fermionic nature of the ADAPT-VQE Ansatz. This is because, in general, the operator $e^{i\theta P_i}$ does not preserve important symmetries, such as the particle number, total spin, or the z-component of the spin.

Nevertheless, numerical simulation [55] show that qubit-ADAPT-VQE is able to reduce the circuit depth and measurment cost significantly in several instances of molecular systems. In fact, it is observed that when using the Jordan-Wigner encoding, discarding the Z-operators from each Pauli string does not result in a substantial loss in accuracy. This further reduces the effective circuit depth. On the other hand, it is not clear whether this approach can produce similarly high accuracy results for larger systems, where the non-fermionic nature of the Ansatz may have a larger impact.

4 VQE in First Quantization

In the scientific literature, the terms "VQE" and "VQE in Second Quantization" are almost always used synonymously. This is to be expected, considering that the algorithm is based on a mapping between fermions and qubits, which is inherently second quantized. If we look more closely at each step of the algorithm (see Fig. (4)), however, we notice that there is no actual requirement to use Second Quantization. In fact, it is perfectly reasonable to assume that VQE still works if we use a First Quantization approach, at least in principle. There are now two immediate questions we should ask: (i) is it possible to implement the steps of the VQE algorithm efficiently in First Quantization? and (ii) is there any advantage of using this framework? It turns out that both questions are far from obvious and quite difficult to answer.

At the time of writing, we are not aware of any research exploring concrete implementations of VQE in First Quantization. Second Quantization methods, on the other hand, have enjoyed almost a decade of research and are thus increasingly optimized with regard to the number of qubits, gate complexity, and number of measurements. The aim of this chapter is to take a first step in a new direction and introduce one working implementation of the algorithm in First Quantization from start to finish.

We begin by introducing the encoding scheme that is typically used to map fermionic wavefunctions to qubits and show how to transform the molecular Hamiltonian accordingly. Next, we explore an efficient antisymmetrization circuit based on sorting networks that is typically used for different quantum chemistry simulation but is equally applicable to VQE. In sections 4.3 and 4.4, we then introduce two possible measurement strategies and highlight advantages and disadvantages compared to Second Quantization approaches.

4.1 Encoding the Wavefunction and Hamiltonian

In the typical Second Quantization approach to VQE, the antisymmetry of fermionic many-particle states is enforced through the anti-commutation relations of the creation and annihilation operators $a_i^\dagger$ and a_i. A fermion-to-qubit encoding is then chosen such that the transformed qubit operators s_i^+ and s_i^- satisfy the same relations and thus represent the same state on a quantum computer. A consequence of this method is that a state of N fermionic modes always requires N qubits, regardless of the number of electrons. Thus, even if we are only interested in the state of m electrons, we necessarily have to work in a space that is large enough to represent N electrons. Since we generally assume that $N > m$, we might wonder if it is possible to restrict the Hilbert space to only include m electron states.

In First Quantization, we work with the fermionic m particle wavefunction directly and enforce the antisymmetry of the state by arranging single-electron basis functions in the form of a Slater determinant. In the following, we consider a set of N basis functions $|\phi_i\rangle$.

In order to encode the m electron wavefunction on a quantum computer, we begin by identifying each basis function $|\phi_i\rangle$ with its index i. This number can then be expressed as a binary string $\text{bin}(i) = i_1...i_n$, using $n = \log_2 N$ bits of information. By identifying the bit i_k with the state of the k-th qubit $|q_k\rangle := |i_k\rangle$ we can therefore encode each basis function $|\phi_i\rangle$ using n qubits:

$$|\phi_i\rangle \mapsto |\text{bin}(i)\rangle = |i_1...i_n\rangle \ . \tag{73}$$

By applying this method to each electron, we are able to encode arbitrary product states of m basis functions using $m \log_2 N$ qubits. In order to map this product state to the corresponding Slater determinant, we apply an antisymmetrization circuit, which we will explore in detail in the next section.
The encoding (73) has several interesting consequences. First, we automatically restrict ourselves to states consisting of exactly m electrons, meaning that the Hilbert space is no longer larger than we need it to be. Second, there is a considerable difference between the scaling $\mathcal{O}(N)$ and $\mathcal{O}(m \log_2 N)$. Consider, for instance, a situation where we are interested in describing a small molecule with very high accuracy. If we are able to keep m relatively small, the logarithmic scaling in N allows us to use a far higher number of basis functions for the same number of qubits, compared to Second Quantization.. Naturally, this is not true in the general case of arbitrary m, but the First Quantization approach allows us to treat the contributions of m and N to the scaling separately and thus allocate our resources more efficiently.
As established in section 2.3.2, the first quantized molecular Hamiltonian can be written in a discretized form as

$$H = \sum_{i=1}^{m} \sum_{p,q=1}^{N} h_{pq} \left|\phi_p\right\rangle_i \left\langle\phi_q\right|_i + \frac{1}{2} \sum_{i \neq j}^{m} \sum_{p,q,r,s=1}^{N} h_{pqrs} \left|\phi_p\right\rangle_i \left|\phi_r\right\rangle_j \left\langle\phi_q\right|_i \left\langle\phi_s\right|_j \ . \tag{74}$$

The next step is to apply the binary encoding transformation as described above and rewrite H in terms of Pauli strings. To illustrate this, consider the case of $N = 4$ basis functions. The orbitals are then mapped to states of $n = 2$ qubits in the following way:

$$\begin{aligned} |\phi_0\rangle &\mapsto |00\rangle \qquad & |\phi_1\rangle &\mapsto |01\rangle \\ |\phi_2\rangle &\mapsto |10\rangle \qquad & |\phi_3\rangle &\mapsto |11\rangle \end{aligned}$$

Each term of the form $|\phi_p\rangle\langle\phi_q|$ in the Hamiltonian can then be written as a tensor product of n single qubit operators $|i\rangle\langle j|$. For instance, the term $|\phi_2\rangle\langle\phi_1|$ in our example becomes

$$|\phi_2\rangle\langle\phi_1| = |10\rangle\langle 01| = |1\rangle\langle 0| \otimes |0\rangle\langle 1| \ . \tag{75}$$

It is straightforward to express each single qubit operator in terms of two Pauli strings. In particular,

$$\begin{aligned} |0\rangle\langle 0| &= \frac{1}{2}(\mathbb{1} + Z) \\ |0\rangle\langle 1| &= \frac{1}{2}(X + iY) \\ |1\rangle\langle 0| &= \frac{1}{2}(X - iY) \\ |1\rangle\langle 1| &= \frac{1}{2}(\mathbb{1} - Z) \end{aligned} \tag{76}$$

Using these expressions we can rewrite equation (75).

$$\begin{aligned} |\phi_2\rangle\langle\phi_1| &= |1\rangle\langle 0| \otimes |0\rangle\langle 1| \\ &= \frac{1}{4}(X + iY) \otimes (X - iY) \\ &= \frac{1}{4}(X \otimes X + iY \otimes X - iX \otimes Y + Y \otimes Y) \end{aligned} \tag{77}$$

Each term $|\phi_p\rangle\langle\phi_q|$ thus results in a sum of $2^n = N$ Pauli strings. Analogously, each term $|\phi_p\rangle|\phi_r\rangle\langle\phi_q|\langle\phi_s|$ can be expressed as a sum of $2^{2n} = N^2$ Pauli strings. Considering equation (74), it is clear that the there are $\mathcal{O}(m^2N^4)$ such terms in the Hamiltonian and it might therefore seems like the total number of Pauli strings in the Hamiltonian scales as $\mathcal{O}(m^2N^6)$. As it turns out, however, by expanding the Hamiltonian in this way, we have ignored the fact that the same Pauli string may appear more than once. To estimate the scaling more precisely, we notice that for each single electron register there are only $4^n = N^2$ possible different Pauli strings. Similarly, for every pair of electrons there are $4^{2n} = N^4$ possible Pauli strings. It must therefore be possible to expand the Hamiltonian as a sum of $\mathcal{O}(m^2N^4)$ Pauli strings. In practice, this may be achieved by generating a list of all possible Pauli strings and then computing the correct prefactors. The prefactor λ_i to the Pauli string P_i is computed by finding all occurrences of P_i in the naive expansion of H and summing up their prefactors.

4.2 Antisymmetrization

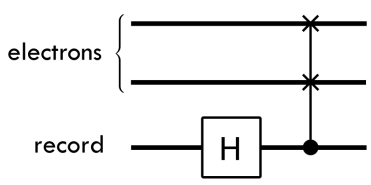
In section 4.1 we have seen how to prepare an arbitrary product state of m single electron orbitals $|\phi_i\rangle$. The next step is to antisymmetrize this state, i.e. perform the transformation

$$|\phi_1\rangle \otimes ... \otimes |\phi_m\rangle \mapsto |\phi_1 ... \phi_m\rangle := \frac{1}{\sqrt{m!}} \sum_{\sigma \in S_m} (-1)^\sigma \left|\phi_{\sigma(1)}\right\rangle \otimes ... \otimes \left|\phi_{\sigma(m)}\right\rangle . \tag{78}$$

An implementation of this transformation on a quantum computer was first proposed by Lloyd and Adams [2] in the context of quantum chemistry simulations using a phase estimation algorithm. Their method results in the antisymmetrization of arbitrary, sorted input states by means of a circuit of gate depth $\mathcal{O}(m^2 \log^2 N)$.
Recently, Babbush et al. introduced an improved method [13], achieving the same goal with circuit depth $\mathcal{O}(\log^c m \log\log N)$, where $c \geq 1$. This significant reduction in complexity allows for the state preparation to be a relatively cheap step in quantum chemistry simulations, which does not contribute much to the overall resource scaling.
The aim of this chapter is to give a detailed outline of the method proposed by Babbush et al in [13]. As some of steps of the algorithm can be unintuitive at first, we begin by providing some motivation of the key concepts. In particular, we outline how one might set up the antisymmetrization of a simple example case, in order to highlight some of the challenges and motivate the choices of the algorithm.

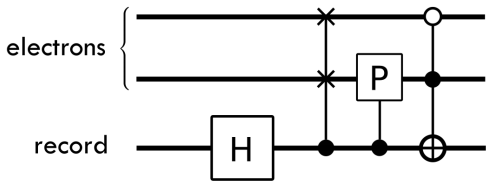
4.2.1 Motivation

Looking at equation (78), the effect of the antisymmetrization circuit is to generate a superposition of all possible permutations of the input product state $|\psi_{in}\rangle$, while including a factor of -1 for every odd permutation. The key to achieving such a superposition is to introduce a set of ancilla qubits, place them in a superposition state that is easier to generate, and then apply a set of controlled swap operations on the electron registers, conditioned on the ancillas. For reasons that will become clear later, we will refer to this ancilla register as `record`.
To illustrate this, consider the simplest non-trivial situation, namely, a state $m = 2$ electrons and $N = 2$ basis functions, together with a single ancilla qubit. We initialize the electrons in the product state $|10\rangle$ and the ancilla in the superposition state $(|0\rangle + |1\rangle)/\sqrt{2}$ by applying a Hadamard gate. In order to transfer this superposition to the electrons, we apply a CSWAP-gate with the ancilla as control qubit and the electrons as target qubits.



$$|10\rangle |0\rangle \mapsto \frac{1}{\sqrt{2}}\Big(|10\rangle |0\rangle + |01\rangle |1\rangle\Big) \tag{79}$$

Evidently, this does not produce the correct transformation yet. First, we need to include the correct sign of the permutation by including a controlled phase on one of the electrons. Second, and most importantly, the state of the electrons is now entangled with the state of the ancilla. Simply discarding the ancilla qubit would thus collapse the electrons onto a mixed state. An essential step of the algorithm is to disentangle the electrons from the ancilla, which can be achieved in different ways. In our example, we can simply add a controlled X-gate on the ancilla, conditioned on the first electron begin in the state $|0\rangle$ and the second electron begin in the state $|1\rangle$.



$$|10\rangle\,|0\rangle \mapsto \frac{1}{\sqrt{2}}\Big(|10\rangle - |01\rangle\Big)\,|0\rangle \tag{80}$$

We may now discard the ancilla qubit and are left with a pure antisymmetrized version of our input state.

The algorithm by Babbush et al. is essentially a generalization of this simple case to an arbitrary number of electrons m and orbitals N. There are a few important differences, however, which lead to the individual steps being more complex.

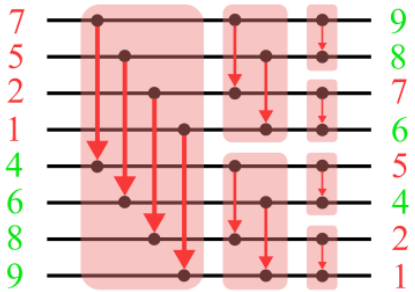
For example, if $m \geq 3$, we need a more complex ancilla register and it is no longer sufficient to simply generate superposition states using Hadamard gates. To see this, consider an ancilla register of L qubits and apply a Hadamard gate to each. The resulting superposition consists of 2^L unique states. On the other hand, the number of terms in an antisymmetrized m-electron state is equal to $m!$. Since $2^L \neq m!$ in general, such a register cannot be used in combination with simple CSWAP gates to generate the correct state in a straightforward way. A key result of the paper by Babbush et al. is that you can solve this problem by preparing a suitable ancilla register using sorting networks. The inclusion of sorting networks has the consequence that the algorithm is no longer deterministic. More specifically, it becomes necessary to include a "Repetition-Check", which can fail with a certain probability. Babbush et al showed, however, that by choosing certain parameters, it is always possible to keep the probability of success close to 1. In the context of VQE, this means that we may have to repeat the state preparation step but are guaranteed a successful state after a small number of repetitions.

4.2.2 Sorting Networks

Sorting networks are logical circuits that perform the task of sorting an input array consisting of a predetermined and fixed number of elements. In the following, we denote the number of elements by m and choose the convention of sorting in descending order from the top. A sorting network is characterized by m horizontal wires carrying values from left to right and a set of comparator-swaps, each acting on one pair of wires. The characterizing feature of sorting networks is that the locations of these comparator-swaps are predetermined and independent of the input. Each comparator-swap takes as input two numbers a and b and performs a swap either if $a > b$ (indicated by a red arrow) or if $b > a$ (indicated by a green arrow).

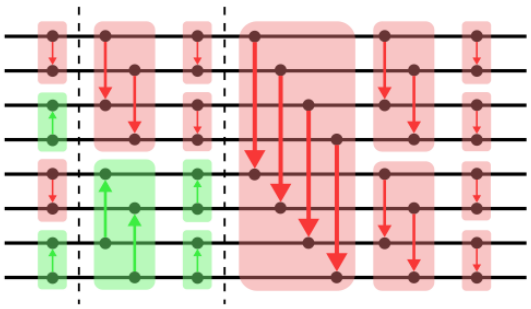
There are many different types of sorting networks that differ in the required number of comparators. The asymptotically best performance is achieved by networks that have depth $\mathcal{O}(\log m)$, however, they typically include a large prefactor [6] and thus become inefficient for realistic m. In the following, we will focus on a particular instance called a *bitonic sorting network* [9], as it exhibits a gate depth scaling of $\mathcal{O}(\log^2 m)$ and allows efficient representations for small m .

A sequence of numbers is defined to be *bitonic* if it consists of a single non-increasing sequence, followed by a single non-decreasing one of equal length, or vice-versa. Bitonic sorting networks are built on the fact that we can sort bitonic sequences by applying the following network of comparators, here shown for the case $m = 8$ and an example sequence.



Each element in the top half is compared to its corresponding element in the lower half, moving each number in the correct half of the total sequence. The same procedure is then repeated for smaller blocks of half the size. If the input happens to be a bitonic sequence,

the output is completely sorted. We can now construct the full bitonic sorting network by sorting bitonic sequences of increasing size. We begin with comparators on every adjacent pair of input elements to construct bitonic sequences of length 4. Then, the first sequence is sorted in non-increasing order, the second one in non-decreasing order, and so on. This process is repeated until we are left with one bitonic sequence that can be transformed into the sorted output.



From this construction it is clear that the number of parallel comparison rounds in the network is $\lceil \log m \rceil (\lceil \log m \rceil + 1)/2$. In the example above, there is one round of comparisons in the first layer, two in the second layer, and three in the third. Since each round can have a maximum of $\lfloor m/2 \rfloor$ comparators, we can bound the total number of comparators L by

$$L \leq \lfloor m/2 \rfloor \cdot \lceil \log m \rceil (\lceil \log m \rceil + 1)/2 \,. \tag{81}$$

4.2.3 Main Algorithm

The algorithm [13] consists of four main steps. In step 1, we prepare a qubit register of size mn' called `seed` and apply a Hadamard gate to each qubit. In step 2, we apply a sorting network to the state of `seed` in order to map each element in the superposition to a sorted state. Additionally, we store the outcome of each comparator in a second register called `record`, making the sorting network reversible. In step 3, since the sorted state of `seed` may contain repetitions, we need to project it onto the repetition free subspace. This can be achieved by the measurement of a single ancilla qubit and the probability of success is determined by the choice of n' in step 1. If successful, the states of `record` and `seed` are disentangled and we may discard `seed` but keep `record`. In step 4, we prepare a product state in a third register `target` of size mn and apply the reverse of the sorting network, again using `record` as control qubits. Due to the implementation of the sorting network, we may now discard `record`, as it is disentangled from `target`.
The algorithm can be implemented using $\mathcal{O}(m \log^c m \log N)$ elementary gates and circuit depth $\mathcal{O}(\log^c m \log \log N)$ and the number of ancilla qubits scales as $\mathcal{O}(m \log^c m \log N)$. In particular, $c = 2$ for bitonic sorting networks.

Step 1: Prepare `seed`

In the final implementation of VQE, each electron is represented by n qubits. The register `seed` mirrors this construction, except that we choose n' qubits instead of n. The choice of n' is independent of n and will later allow us to influence the probability of success in step 3 of the algorithm. We initialize all qubits in `seed` in the state $|0\rangle$ and subsequently apply a Hadamard gate to each one. This results in a superposition of all length-m strings of numbers between 0 and $2^{n'} - 1$.

$$|0\rangle^{\otimes mn'} \mapsto \frac{1}{\sqrt{2^{mn'}}} \sum_{s_1 \ldots s_m}^{2^{n'}-1} |s_1 \ldots s_m\rangle \quad \text{where} \quad 0 \leq s_i < 2^{n'} \tag{82}$$

Step 2: Sort `seed`

Having prepared the state of `seed`, we now apply a sorting network to sort all element in the superposition (82). More specifically, each set of n' qubits represents one wire of a sorting network of size m. One may also think of the wires as representing the individual electrons. In order to use a sorting network as introduced in section 4.2.2 on a quantum computer, we first need a way of implementing comparators in a reversible way on qubits. It is essential to specify the criterion *reversible*, since we want to store the information whether each comparator performed a swap. For every comparator, each acting on two wires comprised of $2n'$ qubits, we thus need one ancilla qubit. If we have chosen a sorting network consisting of L comparators, the register `record` must also contain L qubits. Each comparator is constructed by setting the ancilla qubit to $|1\rangle$ if and only if the inputs a and b satisfy the condition $a < b$. A CSWAP-gate, conditioned on the state of the ancilla, then performs swap if and only if $a < b$ is satisfied.

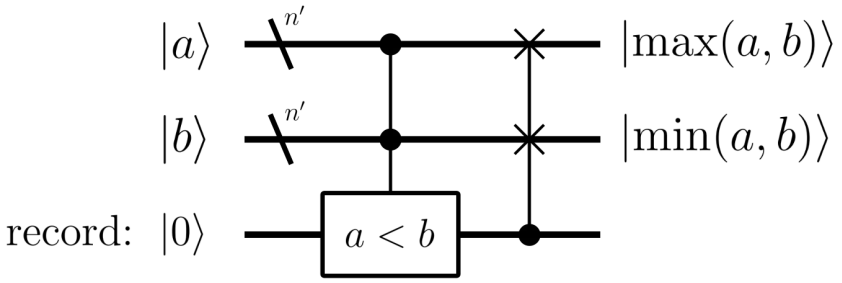


FIGURE 6: Circuit implementation of the reversible comparator. It takes two n'-bit numbers a and b as input and performs a swap iff. $a < b$. The information whether a swap was performed is stored in an additional ancilla qubit.

The first part of this circuit can be implemented in two steps. The idea is to transform the state of $2n'$ qubits in such a way that we end up with two qubits q_1 and q_2 which have the same relation with each other as a and b, i.e. $\text{sgn}(a-b) = \text{sgn}(q_1 - q_2)$. After that it is a simple step to apply an X-gate to the ancilla qubit if q_1 and q_2 are in the joint state $|01\rangle$. The key building block of this method is the circuit depicted in Fig. 7, which we will refer to as `Compare-2`. It takes as input two 2-bit numbers $a = a_0 2^0 + a_1 2^1$ and $b = b_0 2^0 + b_1 2^1$ and generates as output two single bits a' and b', satisfying $\text{sgn}(a - b) = \text{sgn}(a' - b')$ and therefore preserving the inequalities. The positions of a' and b' always coincide with the positions of the second bit in each number, i.e. a_1 and b_1. The outputs at the positions of a_0, b_0, and the ancilla are labeled "temp", as they are not needed for further computations. In particular, once the comparison of registers a and b is performed and the result is written to a `record` qubit, all instances of `Compare-2` are applied in reverse, such that "temp" is uncomputed and the input is reconstructed.

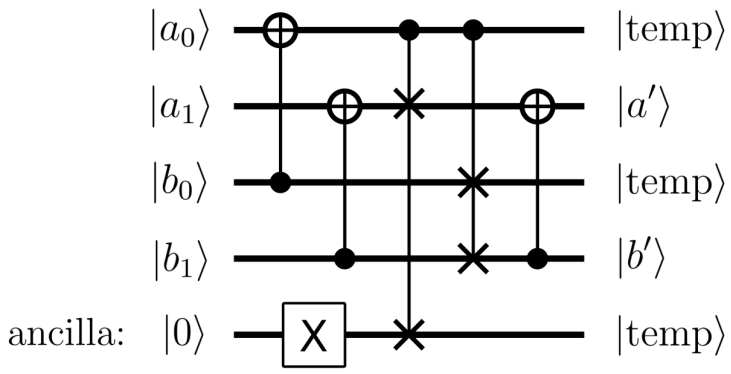


FIGURE 7: Circuit implementation of the function `Compare-2`, taking as input the 2-bit numbers $a = a_0 2^0 + a_1 2^1$ and $b = b_0 2^0 + b_1 2^1$ and generating as output the 1-bit numbers a' and b', such that $\text{sgn}(a - b) = \text{sgn}(a' - b')$.

`Compare-2` reduces the problem of comparing two 2-bit numbers to comparing two 1-bit numbers. The question now is how to use this circuit to achieve the comparison of two n'-bit numbers. As it turns out, we can solve this problem efficiently by using parallelized bitwise comparisons.

Consider first the case that n' is a power of 2. The algorithm can later be generalized for arbitrary n' in a straightforward way. We begin by splitting the array a into $n'/2$ slices, such that a_0 and a_1 are in one slice, a_2 and a_3 in another, and so on. Next, we split b in exactly the same way and perform the following comparisons. First, compare the first slice of a with the first slice of b, which is achieved by applying `Compare-2` with inputs (a_0,a_1,b_0,b_1) and overwriting the second entry of both slices. Then, compare the second slices by applying `Compare-2` with inputs (a_2,a_3,b_2,b_3), and so on until the last two slices. Note that all $n'/2$ instances of `Compare-2` can be applied in parallel, with a requirement of $n'/2$ ancillas. We are left with two $n'/2$-bit numbers a' and b', whose bits are located at the positions that were previously occupied by $a_1, a_3, ..., a_{n'-1}$ and $b_1, b_3, ..., b_{n'-1}$. Now, we repeat the exact same procedure using the numbers a' and b' as input, further reducing

the number of bits to $n'/4$ and introducing an additional $n'/4$ ancilla qubits. After $\log_2 n'$ repetitions we have reached our goal of reducing the comparison between a and b to the comparison of two qubits q_1 and q_2 at positions $a_{n'-1}$ and $b_{n'-1}$. In summary, we have used $n'/2 + n'/4 + n'/8 + ... + 1 = n' - 1$ ancilla qubits.
If n' is not a power of 2, there will be at least one iteration in the procedure where it is not possible to split the entire array into slices of size 2. In that case, we simply keep the remaining pair of numbers until the next iteration without involving them in any comparison. An example of the parallelized bitwise comparison is depicted in Fig. 8 for the case of $n' = 9$.

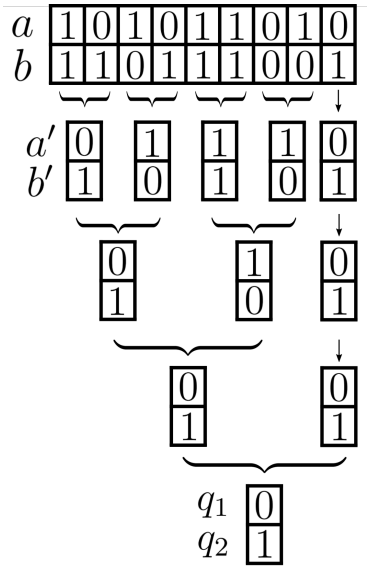


FIGURE 8: Example of the parallelized bitwise comparison of two 9-bit numbers $a = 346$ and $b = 441$. Each curly bracket represents the application of `Compare-2` to two bits from the binary representation of a and two from the binary representation of b. The output consists of two bits q_1 and q_2, satisfying the relation $\mathrm{sgn}(a - b) = \mathrm{sgn}(q_1 - q_2)$.

The next step is to perform a final comparison of the two qubits q_1 and q_2 and store the outcome in a single ancilla qubit in the `record` register. This qubit is now in the state $|1\rangle$ if $a < b$ and in the state $|0\rangle$ if $a \geq b$.

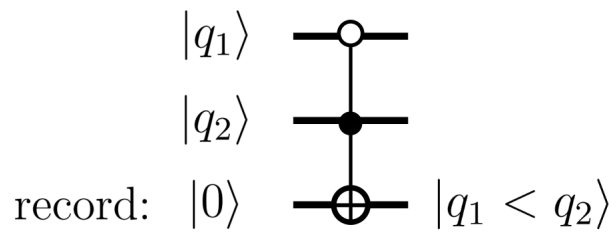


FIGURE 9: Circuit implementation of a single bit comparison. An ancilla qubit is set to $|0\rangle$ iff. $q_1 < q_2$.

At this point, we have successfully compared the registers a and b and written the result to a `record` qubit. However, in doing so we have changed the state of the registers as

indicated by the label "temp" in Fig. 7, such that we can no longer directly access the numbers a and b. In order to reconstruct the input, we keep a list of all gates that have been applied prior to writing to the `record` qubit and apply them in reverse order.
The next step is to swap the registers a and b, conditioned on the state of the qubit in `record`. Since each register contains n' qubits, we can simply apply n' CSWAP gates in sequence and exchange the bits in the binary representations individually (swap a_0 and b_0, swap a_1 and b_1, etc.).
Using the reversible comparator, it is straightforward to apply a sorting network to the full system. An essential feature of the bitonic sorting network is that $m/2$ comparators can be applied in parallel. Since each comparator requires $n'-1$ additional ancilla qubits, it is enough to use $m/2 \cdot (n'-1)$ ancillas and and reuse them for the comparators in the next layer.

Step 3: Delete Repetitions from `seed`

After applying a sorting network to the state (82), we may write the joint state of `seed` and `record` in the following way.

$$\frac{1}{\sqrt{2^{mn'}}} \sum_{s_1 \geq ... \geq s_m}^{2^{n'}-1} |s_1...s_m\rangle_{\texttt{seed}} \, |\psi(s_1, ..., s_m)\rangle_{\texttt{record}} \tag{83}$$

There are two immediate problems with this. First, even though the state of `seed` is sorted, it is still entangled with the state of `record`. Second, `seed` contains both states without repetitions ($s_i \neq s_k \; \forall i \neq k$), and states with repetitions ($\exists \, i, k : \; s_i = s_k$). It turns out that we can solve both problems elegantly by projecting the state of `seed` onto the repetition-free subspace. We define the repetition-free subspace $\mathcal{F} := \text{span}\{|s_1...s_m\rangle \, | s_i \neq s_k \; \forall \, i \neq k\}$ and its orthogonal complement $\mathcal{F}^\perp$, such that $\mathcal{H}_{\texttt{seed}} = \mathcal{F} \cup \mathcal{F}^\perp$.
The projection onto $\mathcal{F}$ can be implemented using the components we have already defined. The idea is to set an ancilla qubit to $|1\rangle$ if and only if the state of m electrons in `seed` is repetition-free. A measurement of this qubit then performs the projection on $\mathcal{F}$ with probability p_{succ} and on $\mathcal{F}^\perp$ with probability $1-p_{\text{succ}}$. Since the sorting network is unitary and does not change the occurrence of repetitions, it does not matter whether the projection is performed before or after sorting. However, sorting `seed` first allows us to only compare adjacent registers, instead of comparing all pairs. We therefore apply a modified reversible comparator to the registers i and $i+1$, for all $i = 0, ..., m$. The comparator differs from the one we used in the sorting network in two ways. First, instead of writing the result of the comparison to a qubit in `record`, we introduce a new ancilla register `repetition` consisting of $m-1$ ancillas. Second, we need to set the state of the ancilla qubit to $|1\rangle$ if the two input registers a and b satisfy $a \neq b$. Since we are guaranteed that $a \geq b$, we only need to test whether $a > b$, which can be done by applying a controlled gate

as in Fig. 9 but changing the control-state from $|01\rangle$ to $|10\rangle$. Note that all comparisons of adjacent registers can be applied in two parallelized layers by first comparing registers i and $i+1$ for all even indices in the first layer and then for all odd indices in the second layer.

At this point, the state of `seed` is repetition-free, if all qubits in `repetition` are set to $|1\rangle$. We can either measure all qubits in `repetition` directly or transfer this information to a single ancilla qubit by applying a controlled-X gate, conditioned on the state $|1\rangle^{\otimes(m-1)}$. Whereas measuring all qubits minimizes the cost of additional qubits and gate depth, measuring a single qubit minimizes the potential read-out error from measurements.

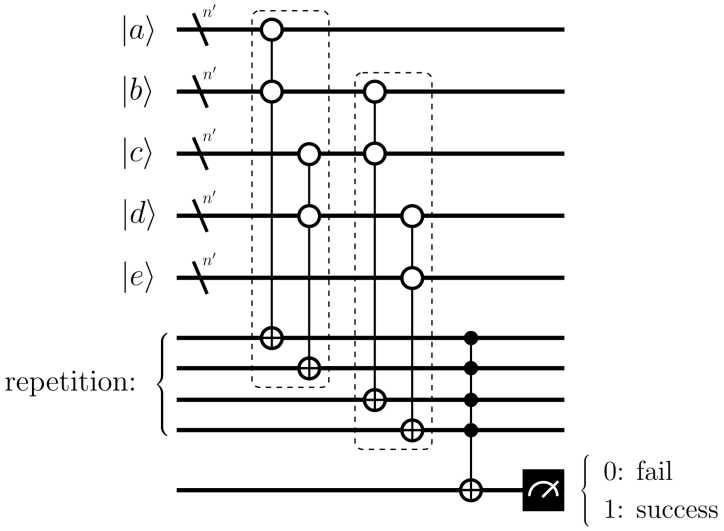


FIGURE 10: Circuit implementation of a repetition check. An ancilla qubit is set to $|1\rangle$ iff. all registers are different and no repetitions are present. All comparisons in the dashed boxed can be executed in parallel.

The probability of success p_{succ} is the probability that a measurement of the ancilla qubit in Fig. 10 returns $|1\rangle$. It can be determined by projecting the state (82) onto the repetition-free subspace and calculating norm of the resulting vector (see Appendix A.2). Then, the probability of success can be shown to be bounded by

$$p_{\text{succ}} \geq 1 - \frac{m(m-1)}{2^{n'+1}}\,. \tag{84}$$

In particular, choosing n' such that $2^{n'} \geq m(m-1)$ guarantees that $p_{\text{succ}} \geq 1/2$. We thus obtain the repetition-free outcome with fewer than two tries on average. A crucial consequence of a successful projection onto $\mathcal{F}$ is that `seed` and `record` are no longer entangled (see Appendix A.3). We have therefore prepared the correct pure state in `record` and can discard `seed`.

Step 4: Apply the reverse of the sorting network to `target`

The final step of the algorithm is the actual antisymmetrization of a target state. The register `target` consists of mn qubits which represent the electrons in the VQE simulation. At this point, `seed` and `repetition` are not used anymore, so the qubits in those registers can be reset and used to build the register `target`. In the simplest case, if $n' = n$, we can simply use `seed`. First, `target` is initialized in a product state corresponding to the result of a classical simulation, typically given by a Hartree-Fock state. Next, the same sorting network that was used to sort `seed` is applied in reverse. The only necessary change is to include a conditional phase gate for each comparator, which has the effect of making the final state antisymmetric, as opposed to symmetric. Each comparator in Fig. 6 is reversed such that we first perform a CSWAP-gate, conditioned on the state of `record`. Then we apply a conditional phase gate with `record` as the control qubit and any one of the qubits in `target` as the target qubit. It does not matter which qubit is chosen, as the phase multiplies each term in the superposition by -1. Finally, an X-gate is applied to `record` if $a < b$ is satisfied. This has the effect of resetting `record` to the state $|0\rangle^{\otimes L}$, such that it can be discarded at the end and the final state in `target` is a pure antisymmetric state.

Fig. 11 shows an outline of the entire algorithm for the case of $m = 3$ electrons. For clarity we have omitted the $n' - 1$ ancilla qubits necessary for each application of the sorting network.

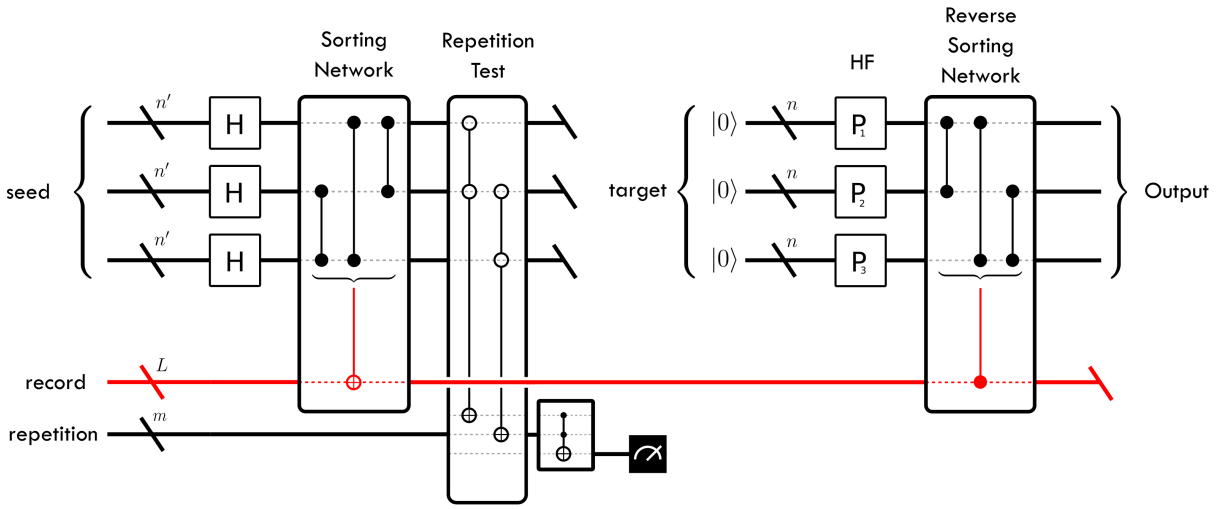


FIGURE 11: Overiew of the Antisymmetrization Circuit as proposed in [13] for the example of $m = 3$. In step 1, `seed` is prepared in a superposition of all length-m strings of numbers between 0 and $2^{n'} - 1$. In step 2, `seed` is sorted by applying a sorting network and storing the outcomes of each comparator in `record`. In step 3, the resulting state is projected onto the repetition free subspace with probability p_{succ}, preparing the correct state in `record` and disentangling it from `seed`. In step 4, a product state `target` is prepared and the sorting network is applied in reverse with the effect of antisymmetrizing target. For clarity, the ancillas necessary for applying the sorting network are omitted.

4.3 Plane Wave Approach

One of the major challenges in developing efficient implementations of VQE is that the molecular Hamiltonian H has up to $\mathcal{O}(m^2N^4)$ terms. Since estimating the expectation value of H requires measuring each of these terms individually, the total number of measurements presents a significant obstacle. There are several methods for reducing the amount of measurements, such as distributing the total number of measurements efficiently among the terms or measuring commuting sets of Pauli strings simultaneously. The approach we will consider in this chapter is to transform the Hamiltonian into a different basis, such that it can be represented using fewer elements.
In a recently published paper [8], Babbush et al proposed an algorithm for quantum chemistry simulations using the plane wave basis. They showed that working in a Second Quantization framework, such a basis change reduces the number of elements in the Hamiltonian from $\mathcal{O}(N^4)$ to $\mathcal{O}(N^2)$.
The aim of this chapter is to review the plane wave basis and its application to variational quantum algorithms as described in [8]. In particular, we formulate the method in a First Quantization context and highlight whenever there are differences to the Second Quantization approach. Furthermore, we will discuss the difficulties connected with using a plane wave approach in real VQE applications.

4.3.1 The Plane Wave Basis

In the plane wave basis, the single electron basis functions $|\phi_i\rangle$ are represented by plane waves in three-dimensional space. Whereas Gaussian and Slater-type orbitals are localized around the position of the nucleus, plane waves are, by definition, highly delocalized and spread over all space. One may therefore immediately raise the question whether this is a suitable basis choice for describing single molecules. Indeed, in order to reach the same level of accuracy in approximating the electronic wavefunction of a molecular system, one would need a far higher number of plane waves, as compared to Gaussian or Slater-Type orbitals [8]. However, since we are working in a First Quantization framework, the number of qubits scales as $\mathcal{O}(m \log N)$, i.e. logarithmically with the number of orbitals, significantly reducing the cost of using a higher number of basis functions. If we are able to also reduce the number of measurements, then using plane waves might be a promising approach to quantum chemistry simulations.
Let Ω be the primitive cell volume of an infinite lattice. For simplicity, we assume the lattice to be cubic but all calculations that depend on the cell shape can by generalized to arbitrary lattice types. In the context of molecular simulations, we place a single molecule in the center of each lattice cell. This is particularly natural when describing periodic systems, but the framework can be used for single molecules as well. In this case, we simply have choose the primitive cell volume Ω large enough that the Coulomb

interaction between different copies of the molecule is negligible. Alternatively, we may introduce an interaction cutoff that is smaller than the cell size.
In order to define a basis consisting of N elements, we generate the reciprocal lattice and choose N points $\boldsymbol{k_p}$ within a cube of length $\frac{2\pi N^{1/3}}{\Omega^{1/3}}$. In particular, we define

$$\boldsymbol{k_p} := \frac{2\pi \boldsymbol{p}}{\Omega^{1/3}}\,, \qquad \boldsymbol{p} \in G := \left[-\frac{N^{1/3}}{2}, \frac{N^{1/3}}{2}\right]^3 \subset \mathbb{Z}^3\,. \tag{85}$$

The plane wave basis is defined as the set $\{|\phi_{\boldsymbol{p}}\rangle\}_{\boldsymbol{p}\in G}$, such that the wavefunctions in real space are given by

$$\phi_{\boldsymbol{p}}(\boldsymbol{r}) = \sqrt{\frac{1}{\Omega}} e^{-i\boldsymbol{k_p r}}\,. \tag{86}$$

In the following, we typically write $|\boldsymbol{p}\rangle := |\phi_{\boldsymbol{p}}\rangle$. The molecular Hamiltonian (25) can be expressed as a sum $H = T + U + V$, where T represents the kinetic energy, U the external potential energy and V the interaction energy. Analogous to equations (26) and (27), we express each term using a set of matrix elements. For simplicity, we use atomic units, such that $\hbar = m_e = 1$.

$$T_{\boldsymbol{pq}} = \int_\Omega \mathrm{d}\boldsymbol{r}\, \phi^*_{\boldsymbol{p}}(\boldsymbol{r}) \left(-\frac{\nabla^2}{2}\right) \phi_{\boldsymbol{q}}(\boldsymbol{r}) \tag{87}$$

$$U_{\boldsymbol{pq}} = \int_\Omega \mathrm{d}\boldsymbol{r}\, \phi^*_{\boldsymbol{p}}(\boldsymbol{r}) \left(-\sum_{l=1}^{M} \frac{Z_l}{|\boldsymbol{r} - \boldsymbol{R}_l|}\right) \phi_{\boldsymbol{q}}(\boldsymbol{r}) \tag{88}$$

$$V_{\boldsymbol{pqrs}} = \int_\Omega \mathrm{d}\boldsymbol{r}_1 \mathrm{d}\boldsymbol{r}_2 \frac{\phi^*_{\boldsymbol{p}}(\boldsymbol{r}_1)\, \phi^*_{\boldsymbol{r}}(\boldsymbol{r}_2)\, \phi_{\boldsymbol{q}}(\boldsymbol{r}_2)\, \phi_{\boldsymbol{s}}(\boldsymbol{r}_1)}{|\boldsymbol{r}_1 - \boldsymbol{r}_2|} \tag{89}$$

It is important to note that due to the periodic nature of the lattice, the integrals are only evaluated within the primitive cell. Consequently, the Coulomb potentials must be treated periodically as well. One of the advantages of the plane wave basis is that the integrals (87) - (88) can be evaluated analytically (for a derivation, see Appendix A.4). As a result, we can rewrite the matrix elements as

$$T_{\boldsymbol{pq}} = \delta_{\boldsymbol{pq}} \frac{\|\boldsymbol{k_p}\|^2}{2}\,, \qquad U_{\boldsymbol{pq}} = \frac{4\pi}{\Omega} \sum_{l=1}^{M} Z_l \frac{e^{i\boldsymbol{k_{q-p}} \boldsymbol{R}_l}}{\|\boldsymbol{k_{p-q}}\|^2} \tag{90}$$

$$V_{\boldsymbol{pqrs}} = \delta_{\boldsymbol{p-s,q-r}} \frac{4\pi}{\Omega \|\boldsymbol{k_{p-q}}\|^2} \tag{91}$$

Note that there would be a singularity in the case $\boldsymbol{k} = 0$. However, it can be shown that when treating charge-neutral systems, the singularities appearing in the terms U and V cancel and only contribute a finite constant that depends on the lattice type (see Appendix F in [41] for a derivation). As expected, the kinetic energy term T is diagonal in this basis, since the plane waves $|p\rangle$ are eigenvectors of the operator T. The form of V_{pqrs} implies that $\boldsymbol{p} - \boldsymbol{q} = \boldsymbol{r} - \boldsymbol{s}$, which can be interpreted as a condition for conservation of momentum, only allowing certain transitions between different plane waves. The two-electron operator V can therefore be written as a sum of $\mathcal{O}(m^2N^3)$ terms, as opposed to $\mathcal{O}(m^2N^4)$ in the general case. Using equations (90) and (91), the different parts of the Hamiltonian can be written as

$$T = \sum_{i=1}^{m} \sum_{\boldsymbol{p} \in G} \frac{\|\boldsymbol{k}_{\boldsymbol{p}}\|^2}{2} |\boldsymbol{p}\rangle \langle \boldsymbol{p}|_i \ , \tag{92}$$

$$U = -\frac{4\pi}{\Omega} \sum_{i=1}^{m} \sum_{l=1}^{M} \sum_{\substack{\boldsymbol{p},\boldsymbol{q} \in G \\ \boldsymbol{p} \neq \boldsymbol{q}}} Z_l \frac{e^{-i\boldsymbol{k}_{\boldsymbol{p}-\boldsymbol{q}}\boldsymbol{R}_l}}{\|\boldsymbol{k}_{\boldsymbol{q}-\boldsymbol{p}}\|^2} |\boldsymbol{p}\rangle \langle \boldsymbol{q}|_i \ , \tag{93}$$

$$V = \frac{2\pi}{\Omega} \sum_{i \neq j}^{m} \sum_{\boldsymbol{p},\boldsymbol{q} \in G} \sum_{\substack{\boldsymbol{\nu} \in G \\ \boldsymbol{\nu} \neq 0 \\ \boldsymbol{p}-\boldsymbol{\nu} \in G \\ \boldsymbol{q}+\boldsymbol{\nu} \in G}} \frac{1}{\|\boldsymbol{k}_{\boldsymbol{\nu}}\|^2} |\boldsymbol{p}\rangle \langle \boldsymbol{p} - \boldsymbol{\nu}|_i |\boldsymbol{q}\rangle \langle \boldsymbol{q} + \boldsymbol{\nu}|_j \ . \tag{94}$$

In the expressions above, the addition of momenta is carried out modulo the maximum momentum.

4.3.2 The Plane Wave Dual Basis

As we have seen, transforming the molecular Hamiltonian into a basis of plane waves diagonalizes the kinetic energy T. A natural question is whether we can similarly diagonalizes the external potential U and interaction V. Intuitively, we may expect this to be possible if we choose a grid of delta functions as basis set, such that each basis vector is associated with a real space coordinate. This approach is typically referred to as *Discrete Variable Representation* (DVR) and has been explored in the context of quantum chemistry simulations ([40], [39]). However, the main shortcoming of DVR is that it necessarily introduces an approximation when describing the kinetic energy operator T. Babbush et al [8] showed that this problem can be solved in the context of quantum chemistry simulations on quantum computers using the *plane wave dual basis*. It is obtained by transforming the plane wave basis using a unitary discrete Fourier transform. Note that this would exactly correspond to a real space grid as in DVR if we consider the plane wave basis in the limit of $N \to \infty$. For a finite N, the resulting basis resembles a smooth

approximation to a discrete grid. In order to define the plane wave dual basis, we write the plane wave (86) as a product of its spatial components

$$\phi_{\boldsymbol{p}}(\boldsymbol{r}) = \phi_{p_x}(x) \cdot \phi_{p_y}(y) \cdot \phi_{p_z}(z)\,. \tag{95}$$

Each component is transformed separately using a unitary discrete Fourier transform in its corresponding spatial direction. The resulting functions are then multiplied together to define the wavefunction of an element $\left|\widetilde{\phi}_{\boldsymbol{p}}\right\rangle$ in the plane wave dual basis. For example, the x-component is given by the transformation

$$\widetilde{\phi}_{p_x}(x) = \sqrt{\frac{1}{N^{1/3}}} \sum_{\nu_x} e^{-2\pi i p_x \nu_x / N^{1/3}} \phi_{\nu_x}(x)\,. \tag{96}$$

The y and z-components are defined analogously. Using some algebraic manipulations, the infinite sum in (96) can be expressed in an analytical form (see Appendix A.5 for a derivation)

$$\widetilde{\phi}_{p_x}(x) = \frac{1}{(\Omega N)^{1/6}} \frac{\sin\left(\frac{\pi \bar{N}}{\Omega^{1/3}} x - \frac{\pi p_x \bar{N}}{N^{1/3}}\right)}{\sin\left(\frac{\pi}{\Omega^{1/3}} x - \frac{\pi p_x}{N^{1/3}}\right)}\,, \tag{97}$$

where $\bar{N} := N^{1/3} - 1/2$. Note that this result differs slightly from [8], as we use a different definition of G in the plane wave basis (85). The full wavefunction of the vector $\left|\widetilde{\phi}_{\boldsymbol{p}}\right\rangle$ can now be expressed as the product

$$\widetilde{\phi}_{\boldsymbol{p}}(\boldsymbol{r}) = \sqrt{\frac{1}{\Omega N}} \cdot \frac{\sin\left(\frac{\pi \bar{N}}{\Omega^{1/3}} x - \frac{\pi p_x \bar{N}}{N^{1/3}}\right)}{\sin\left(\frac{\pi}{\Omega^{1/3}} x - \frac{\pi p_x}{N^{1/3}}\right)} \cdot \frac{\sin\left(\frac{\pi \bar{N}}{\Omega^{1/3}} y - \frac{\pi p_y \bar{N}}{N^{1/3}}\right)}{\sin\left(\frac{\pi}{\Omega^{1/3}} y - \frac{\pi p_y}{N^{1/3}}\right)} \cdot \frac{\sin\left(\frac{\pi \bar{N}}{\Omega^{1/3}} z - \frac{\pi p_z \bar{N}}{N^{1/3}}\right)}{\sin\left(\frac{\pi}{\Omega^{1/3}} z - \frac{\pi p_z}{N^{1/3}}\right)}\,. \tag{98}$$

In this representation, $\widetilde{\phi}_{\boldsymbol{p}}(\boldsymbol{r})$ can be interpreted as a smooth approximation to a discrete lattice with grid points at positions at $\boldsymbol{r}_{\boldsymbol{p}} = \boldsymbol{p}(\Omega/N)^{1/3}$. To demonstrate this, Fig. 12, 13, and 14 show the behaviour of two basis functions in one dimension for three separate values of N. The vertical black lines indicate the boundaries of the primitive lattice cell.

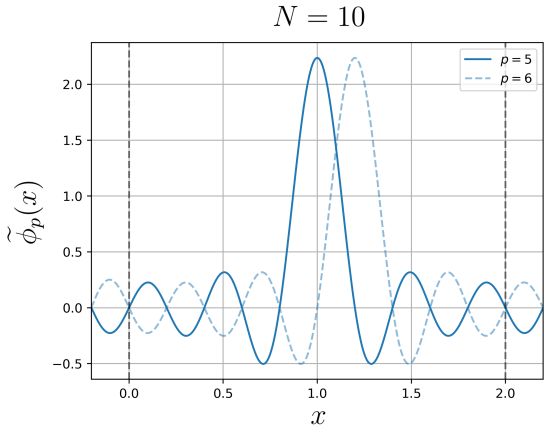


FIGURE 12: Plane wave dual basis $\widetilde{\phi}_{\boldsymbol{p}}(\boldsymbol{r})$ for $N = 10$ in one dimension.

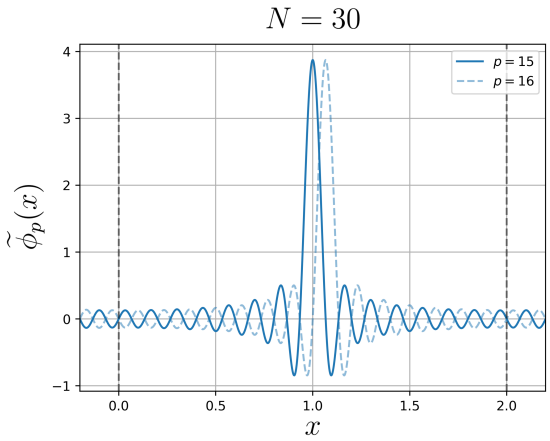


FIGURE 13: Plane wave dual basis $\widetilde{\phi}_{\boldsymbol{p}}(\boldsymbol{r})$ for $N = 30$ in one dimension.

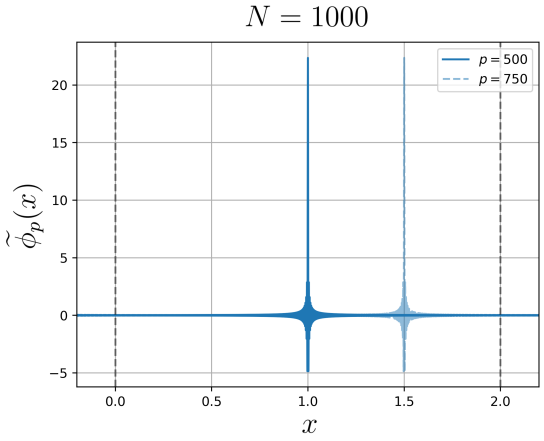


FIGURE 14: Plane wave dual basis $\widetilde{\phi}_{\boldsymbol{p}}(\boldsymbol{r})$ for $N = 1000$ in one dimension.

In the following, we denote basis vectors in the plane wave dual basis by $|\tilde{\boldsymbol{p}}\rangle$. In order to express the Hamiltonian in this basis, we represent the operators T, U, and V in the plane wave basis and express each basis vector using the inverse Fourier transform

$$|\boldsymbol{p}\rangle = \sqrt{\frac{1}{N}} \sum_{\boldsymbol{p}} e^{-i\boldsymbol{k}_{\boldsymbol{p}}\boldsymbol{r}_{\boldsymbol{p}}} |\tilde{\boldsymbol{p}}\rangle \ . \tag{99}$$

Using equation (92), we rewrite the kinetic energy operator

$$T = \sum_{i=1}^{m} \sum_{\boldsymbol{p}} \frac{\|\boldsymbol{k}_{\boldsymbol{p}}\|^2}{2} |\boldsymbol{p}\rangle \langle \boldsymbol{p}|_i \tag{100}$$

$$= \frac{1}{2N} \sum_{i=1}^{m} \sum_{\boldsymbol{p}} \sum_{\tilde{\boldsymbol{p}},\tilde{\boldsymbol{q}}} \|\boldsymbol{k}_{\boldsymbol{p}}\|^2 e^{i\boldsymbol{k}_{\boldsymbol{p}}(\boldsymbol{r}_{\tilde{\boldsymbol{q}}} - \tilde{\boldsymbol{p}})} |\tilde{\boldsymbol{p}}\rangle \langle \tilde{\boldsymbol{q}}| \tag{101}$$

The exponential in (101) can be rewritten to simplify the expression for T. To see this, consider the x-component of $\boldsymbol{p}$ separately:

$$\sum_{p_x=-N/2}^{N/2} k_p^2 e^{ik_p r_{q-p}} = \sum_{p_x=0}^{N/2} k_p^2 \left(e^{ik_p r_{q-p}} + e^{-ik_p r_{q-p}}\right) \tag{102}$$

$$= \sum_{p_x=0}^{N/2} k_p^2 2\cos\left(k_p r_{q-p}\right) \tag{103}$$

$$= \sum_{p_x=-N/2}^{N/2} k_p^2 \cos\left(k_p r_{q-p}\right) \tag{104}$$

where we have used that $k_p^2 = k_{-p}^2$ and $\cos(x) = \cos(-x)$. Using the expression (104), we can write T in the plane wave dual basis as

$$T = \frac{1}{2N} \sum_{i=1}^{m} \sum_{\boldsymbol{p}} \sum_{\tilde{\boldsymbol{p}},\tilde{\boldsymbol{q}}} \|\boldsymbol{k_p}\|^2 \cos\left(\boldsymbol{k_p}(\boldsymbol{r}_{\tilde{\boldsymbol{q}}-\tilde{\boldsymbol{p}}})\right) |\tilde{\boldsymbol{p}}\rangle \langle\tilde{\boldsymbol{q}}| \tag{105}$$

Next, we compute the expression for the external potential U. Using equation (93), we write

$$U = -\frac{4\pi}{\Omega} \sum_{i=1}^{m} \sum_{l=1}^{M} \sum_{\boldsymbol{p}\neq\boldsymbol{q}} Z_l \frac{e^{-i\boldsymbol{k_{p-q}}\boldsymbol{R}_l}}{\|\boldsymbol{k_{q-p}}\|^2} |\boldsymbol{p}\rangle \langle\boldsymbol{q}|_i \tag{106}$$

$$= -\frac{4\pi}{N\Omega} \sum_{i,l} \sum_{\tilde{\boldsymbol{p}},\tilde{\boldsymbol{q}}} \sum_{\boldsymbol{p}\neq\boldsymbol{q}} Z_l e^{i(\boldsymbol{k_q}\boldsymbol{r}_{\tilde{\boldsymbol{q}}} - \boldsymbol{k_p}\boldsymbol{r}_{\tilde{\boldsymbol{p}}})} \frac{e^{-i\boldsymbol{k_{q-p}}\boldsymbol{R}_l}}{\|\boldsymbol{k_{q-p}}\|^2} |\tilde{\boldsymbol{p}}\rangle \langle\tilde{\boldsymbol{q}}|_i \tag{107}$$

$$= -\frac{4\pi}{N\Omega} \sum_{i,l} \sum_{\tilde{\boldsymbol{p}},\tilde{\boldsymbol{q}}} \sum_{\substack{\boldsymbol{q'}\neq 0 \\ \boldsymbol{q}\neq 0}} Z_l e^{i(\boldsymbol{k_q}\boldsymbol{r}_{\tilde{\boldsymbol{q}}-\tilde{\boldsymbol{p}}})} \frac{e^{-i\boldsymbol{k_{q'}}(\boldsymbol{R}_l - \boldsymbol{r}_{\tilde{\boldsymbol{p}}})}}{\|\boldsymbol{k_{q'}}\|^2} |\tilde{\boldsymbol{p}}\rangle \langle\tilde{\boldsymbol{q}}|_i \qquad \text{with } \boldsymbol{q'} := \boldsymbol{q}-\boldsymbol{p} \tag{108}$$

$$= -\frac{4\pi}{N\Omega} \sum_{i,l} \sum_{\tilde{\boldsymbol{p}},\tilde{\boldsymbol{q}}} Z_l \underbrace{\left(\sum_{\boldsymbol{q}\neq 0} e^{i(\boldsymbol{k_q}\boldsymbol{r}_{\tilde{\boldsymbol{q}}-\tilde{\boldsymbol{p}}})}\right)}_{i)} \underbrace{\left(\sum_{\boldsymbol{q'}\neq 0} \frac{e^{-i\boldsymbol{k_{q'}}(\boldsymbol{R}_l - \boldsymbol{r}_{\tilde{\boldsymbol{p}}})}}{\|\boldsymbol{k_{q'}}\|^2}\right)}_{ii)} |\tilde{\boldsymbol{p}}\rangle \langle\tilde{\boldsymbol{q}}|_i \tag{109}$$

The sum $i)$ is only non-zero if $\tilde{\boldsymbol{p}} = \tilde{\boldsymbol{q}}$ and can therefore be expressed as $\delta_{\tilde{\boldsymbol{p}},\tilde{\boldsymbol{q}}} N$. In order to simplify the expression $ii)$, we need to assume that the lattice point $\boldsymbol{q'}$ is contained in the original lattice cell, i.e. $\boldsymbol{q}-\boldsymbol{p} \in G$. This corresponds to carrying out the addition of momenta modulo a maximum value $\boldsymbol{k}_{\max} = 2\pi N^{1/3}/\Omega^{1/3}$. Crucially, this approximation is necessary to derive diagonal expressions for both U and V. It can be shown, however, that the resulting error scales similarly to the discretization error of the Hamiltonian [8]

and can therefore be neglected. As a result, equation (109) can be simplified to

$$U = -\frac{4\pi}{\Omega} \sum_{i,l} \sum_{\tilde{\boldsymbol{p}},\tilde{\boldsymbol{q}}} Z_l \delta_{\tilde{\boldsymbol{p}},\tilde{\boldsymbol{q}}} \sum_{\boldsymbol{q}' \neq 0} \frac{\cos\left(\boldsymbol{k}_{\boldsymbol{q}'}(\boldsymbol{R}_l - \boldsymbol{r}_{\tilde{\boldsymbol{p}}})\right)}{\|\boldsymbol{k}_{\boldsymbol{q}'}\|^2} |\tilde{\boldsymbol{p}}\rangle \langle \tilde{\boldsymbol{q}}|_i \tag{110}$$

$$\Rightarrow U = -\frac{4\pi}{\Omega} \sum_{i,l} \sum_{\tilde{\boldsymbol{p}}} \sum_{\boldsymbol{q}' \neq 0} Z_l \frac{\cos\left(\boldsymbol{k}_{\boldsymbol{q}'}(\boldsymbol{R}_l - \boldsymbol{r}_{\tilde{\boldsymbol{p}}})\right)}{\|\boldsymbol{k}_{\boldsymbol{q}'}\|^2} |\tilde{\boldsymbol{p}}\rangle \langle \tilde{\boldsymbol{p}}|_i \tag{111}$$

A basis change to the plane wave dual basis therefore diagonalizes the external potential operator U.

Finally, we compute the interaction term V in equation (94). Note that the sum over $\boldsymbol{\nu}$ only includes values that satisfy $\boldsymbol{p} - \boldsymbol{\nu} \in G$ and $\boldsymbol{q} + \boldsymbol{\nu} \in G$.

$$V = \frac{2\pi}{\Omega} \sum_{i \neq j} \sum_{\boldsymbol{p},\boldsymbol{q}} \sum_{\boldsymbol{\nu} \neq 0} \frac{1}{\|\boldsymbol{k}_{\boldsymbol{\nu}}\|^2} |\boldsymbol{p}\rangle \langle \boldsymbol{p} - \boldsymbol{\nu}|_i |\boldsymbol{q}\rangle \langle \boldsymbol{q} + \boldsymbol{\nu}|_j \tag{112}$$

$$= \frac{2\pi}{N^2\Omega} \sum_{i \neq j} \sum_{\substack{\boldsymbol{p},\boldsymbol{q},\boldsymbol{\nu} \\ \boldsymbol{\nu} \neq 0}} \sum_{\substack{\tilde{\boldsymbol{p}},\tilde{\boldsymbol{q}} \\ \tilde{\boldsymbol{r}},\tilde{\boldsymbol{s}}}} \frac{1}{\|\boldsymbol{k}_{\boldsymbol{\nu}}\|^2} e^{i(-\boldsymbol{k}_{\boldsymbol{p}}\boldsymbol{r}_{\tilde{\boldsymbol{p}}-\tilde{\boldsymbol{r}}} - \boldsymbol{k}_{\boldsymbol{q}}\boldsymbol{r}_{\tilde{\boldsymbol{q}}-\tilde{\boldsymbol{s}}} - \boldsymbol{k}_{\boldsymbol{\nu}}\boldsymbol{r}_{\tilde{\boldsymbol{r}}-\tilde{\boldsymbol{s}}})} |\tilde{\boldsymbol{p}}\rangle \langle \tilde{\boldsymbol{r}}|_i |\tilde{\boldsymbol{q}}\rangle \langle \tilde{\boldsymbol{s}}|_j \tag{113}$$

$$= \frac{2\pi}{N^2\Omega} \sum_{i \neq j} \sum_{\boldsymbol{\nu} \neq 0} \sum_{\substack{\tilde{\boldsymbol{p}},\tilde{\boldsymbol{q}} \\ \tilde{\boldsymbol{r}},\tilde{\boldsymbol{s}}}} \frac{1}{\|\boldsymbol{k}_{\boldsymbol{\nu}}\|^2} \underbrace{\sum_{\boldsymbol{p}} e^{-i\boldsymbol{k}_{\boldsymbol{p}}\boldsymbol{r}_{\tilde{\boldsymbol{p}}-\tilde{\boldsymbol{r}}}}}_{\delta_{\tilde{\boldsymbol{p}}\tilde{\boldsymbol{r}}} N} \underbrace{\sum_{\boldsymbol{q}} e^{-i\boldsymbol{k}_{\boldsymbol{q}}\boldsymbol{r}_{\tilde{\boldsymbol{q}}-\tilde{\boldsymbol{s}}}}}_{\delta_{\tilde{\boldsymbol{q}}\tilde{\boldsymbol{s}}} N} e^{-i\boldsymbol{k}_{\boldsymbol{\nu}}\boldsymbol{r}_{\tilde{\boldsymbol{r}}-\tilde{\boldsymbol{s}}}} |\tilde{\boldsymbol{p}}\rangle \langle \tilde{\boldsymbol{r}}|_i |\tilde{\boldsymbol{q}}\rangle \langle \tilde{\boldsymbol{s}}|_j \tag{114}$$

$$= \frac{2\pi}{\Omega} \sum_{i \neq j} \sum_{\boldsymbol{\nu} \neq 0} \sum_{\substack{\tilde{\boldsymbol{p}},\tilde{\boldsymbol{q}} \\ \tilde{\boldsymbol{r}},\tilde{\boldsymbol{s}}}} \frac{1}{\|\boldsymbol{k}_{\boldsymbol{\nu}}\|^2} \delta_{\tilde{\boldsymbol{p}}\tilde{\boldsymbol{r}}} \delta_{\tilde{\boldsymbol{q}}\tilde{\boldsymbol{s}}} e^{-i\boldsymbol{k}_{\boldsymbol{\nu}}\boldsymbol{r}_{\tilde{\boldsymbol{r}}-\tilde{\boldsymbol{s}}}} |\tilde{\boldsymbol{p}}\rangle \langle \tilde{\boldsymbol{r}}|_i |\tilde{\boldsymbol{q}}\rangle \langle \tilde{\boldsymbol{s}}|_j \tag{115}$$

$$= \frac{2\pi}{\Omega} \sum_{i \neq j} \sum_{\boldsymbol{\nu} \neq 0} \sum_{\tilde{\boldsymbol{p}},\tilde{\boldsymbol{q}}} \frac{\cos\left(\boldsymbol{k}_{\boldsymbol{\nu}}\boldsymbol{r}_{\tilde{\boldsymbol{p}}-\tilde{\boldsymbol{q}}}\right)}{\|\boldsymbol{k}_{\boldsymbol{\nu}}\|^2} |\tilde{\boldsymbol{p}}\rangle \langle \tilde{\boldsymbol{p}}|_i |\tilde{\boldsymbol{q}}\rangle \langle \tilde{\boldsymbol{q}}|_j \tag{116}$$

The interaction term V can thus be expressed in diagonal form using $\mathcal{O}(m^2N^2)$ terms. To summarize, in the plane wave dual the operators T, U, and V take the following form.

$$T = \frac{1}{2N} \sum_{i=1}^{m} \sum_{\boldsymbol{p}} \sum_{\tilde{\boldsymbol{p}},\tilde{\boldsymbol{q}}} \|\boldsymbol{k}_{\boldsymbol{p}}\|^2 \cos\left(\boldsymbol{k}_{\boldsymbol{p}}(\boldsymbol{r}_{\tilde{\boldsymbol{q}}-\tilde{\boldsymbol{p}}})\right) |\tilde{\boldsymbol{p}}\rangle \langle \tilde{\boldsymbol{q}}| \tag{117}$$

$$U = -\frac{4\pi}{\Omega} \sum_{i,l} \sum_{\tilde{\boldsymbol{p}}} \sum_{\boldsymbol{q}' \neq 0} Z_l \frac{\cos\left(\boldsymbol{k}_{\boldsymbol{q}'}(\boldsymbol{R}_l - \boldsymbol{r}_{\tilde{\boldsymbol{p}}})\right)}{\|\boldsymbol{k}_{\boldsymbol{q}'}\|^2} |\tilde{\boldsymbol{p}}\rangle \langle \tilde{\boldsymbol{p}}|_i \tag{118}$$

$$V = \frac{2\pi}{\Omega} \sum_{i \neq j} \sum_{\boldsymbol{\nu} \neq 0} \sum_{\tilde{\boldsymbol{p}},\tilde{\boldsymbol{q}}} \frac{\cos\left(\boldsymbol{k}_{\boldsymbol{\nu}}\boldsymbol{r}_{\tilde{\boldsymbol{p}}-\tilde{\boldsymbol{q}}}\right)}{\|\boldsymbol{k}_{\boldsymbol{\nu}}\|^2} |\tilde{\boldsymbol{p}}\rangle \langle \tilde{\boldsymbol{p}}|_i |\tilde{\boldsymbol{q}}\rangle \langle \tilde{\boldsymbol{q}}|_j \tag{119}$$

4.3.3 Application to VQE

In each iteration of the VQE algorithm, it is necessary estimate the expectation value $\langle H \rangle$ for the current Ansatz state. In order to sample the observable H on a quantum computer, a fermion-to-qubit mapping is applied to express H as a sum of Pauli strings that can be directly measured. The measurement process can then be optimized by, for example, measuring commuting groups of Pauli strings at the same time. Of course, not all Pauli strings commute and thus each sample of H requires a substantial number of state preparations.
Consider now the case that H is diagonal. We can then express the operator in its diagonal basis as

$$H = \sum_{i,j} \sum_{p,q} h_{pq} \left|p\right\rangle \left\langle p\right|_i \left|q\right\rangle \left\langle q\right|_j \ . \tag{120}$$

As described in section 4.1, we can transform this expression into a sum of Pauli strings using the first quantizied fermion-to-qubit encoding (73). In order to expand the term $|p\rangle \langle q|$, for instance, we first express it as a tensor product of $\log N$ single qubit operators $|i\rangle \langle j|$, where $i, j \in \{0, 1\}$. Each of these operators can be written as the sum of two Pauli matrices as described in (76). Crucially, each term $|p\rangle \langle p|$ can be expanded into a tensor product of the diagonal operators $|0\rangle \langle 0|$ and $|1\rangle \langle 1|$, which in turn can be expressed using the diagonal Pauli matrix Z and the identity $\mathbb{1}$. In summary, the whole operator (120) can be expressed as a sum of Pauli strings that are built from $\mathbb{1}$ and Z only. As a result, all of these Pauli strings commute and can be measured simultaneously, such that H can be sampled once per state preparation.
Evidently, since the entire goal of quantum chemistry simulations is to diagonalize the Hamiltonian, we cannot assume to work in such a basis from the beginning. As we have seen in the previous section, however, we can separately diagonalize the kinetic and potential part of the Hamiltonian by choosing two different bases. More specifically, the plane wave basis diagonalizes the operator T and the plane wave dual basis diagonalizes the operator $U + V$. As originally proposed by Babbush et al in [8], we can therefore sample the expectation value $\langle H \rangle_{\psi_{\boldsymbol{\theta}}}$ in two steps. In step 1, we prepare the Ansatz state $|\psi_{\boldsymbol{\theta}}\rangle$ and measure $U + V$ directly, while evaluating the matrix elements using the plane wave dual basis. In step 2, we prepare $|\psi_{\boldsymbol{\theta}}\rangle$ again, then apply a discrete Fourier transform to perform a basis change, and finally measure T, now evaluating the matrix elements in the plane wave basis. Thus, each circuit repetition enables the sampling of either the complete operator T or the complete operator $U + V$.
Before estimating the scaling of the number of measurements, we consider the cost of applying a discrete Fourier transform. As described in section 2.3.4, the Quantum Fourier Transform on n qubits requires a gate complexity of $\mathcal{O}(n^2)$. At this point, we encounter

the first main difference between First Quantization and Second Quantization approaches. When working in the Second Quantization, the electronic wavefunction is encoded using N qubits, where N is also the number of basis functions. We therefore have to apply the Quantum Fourier Transform to the entire circuit, resulting in a gate complexity $\mathcal{O}(N^2)$. As the gate depth is one of the main limiting factors of VQE, this represents a serious challenge. In fact, the preparation of the Ansatz quickly exhausts the available gate depth and it would best to minimize any additional steps that use this resource.
In a First Quantization approach, on the other hand, this step is significantly cheaper. Since each electron is encoded using $\log N$ qubits, the Quantum Fourier Transform of a single electron only requires a gate complexity $\mathcal{O}(\log^2 N)$. The full electronic wavefunction is then transformed by applying a Quantum Fourier Transform to each of the m electrons simultaneously, resulting in a gate complexity $\mathcal{O}(m \log^2 N)$ and gate depth $\mathcal{O}(\log^2 N)$.

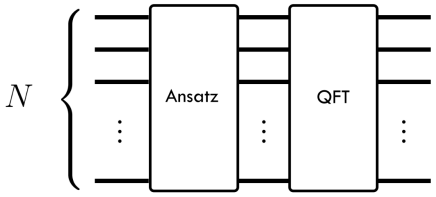


FIGURE 15: Outline of the circuit implementing the basis change using a Quantum Fourier Transform in Second Quantization. The QFT is applied to N qubits.

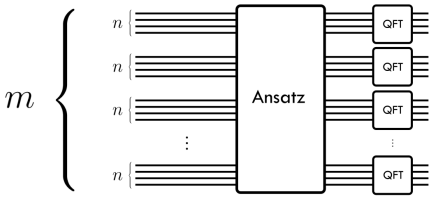


FIGURE 16: Outline of the circuit implementing the basis change using a Quantum Fourier Transform in Second Quantization. The QFT is applied to $n = \log N$ qubits, m times in parallel.

Sampling the Hamiltonian H in this way allows us to define the unbiased estimator $\tilde{H} = \sum_i^M H_i$, where $H_i = T_i + (U + V)_i$ is given by the measurement outcomes of the procedure as described above. Using the standard sampling error $\varepsilon^2 = \mathrm{var}(H)/M$, we can determine the number of measurements M that need to be performed to estimate H up to precision ε. Since T and $U + V$ are sampled using separate copies of the Ansatz state, they can be treated as independent random variables, implying

$$\mathrm{var}(H) = \mathrm{var}(U) + \mathrm{var}(V)\,. \tag{121}$$

The scaling of number of measurements M can thus be expressed as $\mathcal{O}((\mathrm{var}(U)+\mathrm{var}(V))/\varepsilon^2)$. It is shown in [8] that this in turn is equivalent to the following expression.

$$M \in \mathcal{O}\left(\frac{(\max_{\psi} |\langle\psi| T |\psi\rangle|)^2 + (\max_{\psi} |\langle\psi| V |\psi\rangle|)^2}{\varepsilon^2}\right) \tag{122}$$

One can bound the maximum expectation values of T and V (for a derivation, see Appendix A.6).

$$\max_{\psi} |\langle\psi| T |\psi\rangle| \leq C_T \frac{mN^{2/3}}{\Omega^{1/3}} \tag{123}$$

$$\max_{\psi} |\langle\psi| V |\psi\rangle| \leq C_V \frac{m^2 N^{1/3}}{\Omega^{1/3}} \tag{124}$$

Here, C_T and C_V are constant factors that do not depend on m, N, or Ω. Using equation (122), we conclude that the number of measurements M needed to estimate H up to precision ε scales as

$$M \in \mathcal{O}\left(\frac{m^2 N^{4/3}\Omega^{-4/3} + m^4 N^{2/3}\Omega^{-2/3}}{\varepsilon^2}\right). \tag{125}$$

We typically perform quantum chemistry simulations at constant density, i.e. $m/\Omega = \text{const}$. In this case, the scaling relation reduces to

$$M \in \mathcal{O}\left(\frac{m^{2/3} N^{4/3} + m^{10/3} N^{2/3}}{\varepsilon^2}\right). \tag{126}$$

Since in Second Quantization we have $m = \mathcal{O}(N)$, the scaling further simplifies to $M \in \mathcal{O}(N^4/\varepsilon^2)$.

In a First Quantization approach, however, we are able to take advantage of the separation of m and N in the scaling (126). As discussed in section 4.1, performing VQE in an arbitrary basis requires the independent estimation of $\mathcal{O}(m^2N^4)$ Pauli strings. The plane wave approach therefore does not result in a better scaling in both variables m and N. It does, however, significantly reduce the overall scaling with N from $\mathcal{O}(N^4)$ to a worst case of $\mathcal{O}(N^{4/3})$. On the other hand, the scaling with m increases from $\mathcal{O}(m^2)$ to a worst case of $\mathcal{O}(m^{10/3})$. One can certainly imagine situations in which this scaling presents an advantage over Second Quantization methods, such as describing a relatively small molecule with high precision, i.e. using a large number of basis functions. This scenario, which we have already encountered in section 4.1, appears to emerge as a common theme of the potential advantage of First Quantization VQE and represents exactly those situations where Second Quantization approaches are particularly inefficient.

Unfortunately, there are also downsides connected to using the plane wave basis. We have seen that the plane wave basis allows us to estimate the expectation value of the Hamiltonian in a given state using fewer measurements. However, so far we have ignored the state preparation step in this procedure. Typically, the VQE algorithm starts by

initializing the electrons in a state that was obtained by, for example, a Hartree-Fock calculation or any other classical simulation. In HF theory, electron orbitals are described by localized basis functions, such as GTOs or STOs. If we use basis functions of the same kind in our implementation of VQE, then initializing the electron registers in a HF state is a trivial step, as we simply have to apply an X-gate to certain qubits. If we work in the plane wave basis, however, expressing the HF state is an expensive step that requires a significant amount of gates to implement. This behaviour is not entirely surprising, as the choice of basis in VQE typically involves a trade-off between *a.*) the minimization of the number of terms in the Hamiltonian and *b.*) the efficient preparation of a suitable initial state. This is because *a.*) requires a localized basis in momentum space (i.e. plane waves) and *b.*) requires a localized basis in real space (i.e. Gaussian basis), so we typically have to prioritize one over the other [8].
To deal with this problem, we could either perform a classical simulation that already uses the plane wave basis, or we could ignore the initialization step altogether and immediately apply an Ansatz circuit. Both options are possible topics for future research and in order to compare their final performances, it would be beneficial to implement a classical simulation of VQE in First Quantization.

4.4 Adaptive Approach

In the previous section, we have discussed the plane wave basis as a possible means to reduce the number of measurements in each iteration of the VQE routine. However, we have not yet considered what an Ansatz in a First Quantization approach to VQE might look like. The aim of this section is to take those Ansätze that were introduced in section 3.3 and explore if and how they may be implemented in First Quantization.
The first Ansatz that we discussed in section 3.3.1 is the Hardware-Efficient Ansatz (HEA). Although this Ansatz appears to be well motivated by choosing gates that are native to the specific quantum computing hardware that is used for the experiment, it is likely not a suitable approach for larger, near-term devices [57]. This is because it is not physically motivated and thus needs to span a large portion of the Hilbert space, leading to a large number of parameters and a severe barren-plateau problem. As this problem is not specific to the framework of Second Quantization, we have no reason to expect a different performance in First Quantization.
Next, we introduced the Unitary Coupled Cluster Ansatz, which improves over HEA but still requires a considerable circuit depth that is likely to be a strong limiting factor for any near-term applications. We show in Appendix A.7 that translating this approach to First Quantization increases the necessary circuit depth, implying that this approach is probably not suitable.
Instead, we will focus on adaptive structure Ansätze as more promising candidates for an efficient Ansatz in First Quantization VQE.

4.4.1 ADAPT-VQE

The ADAPT-VQE algorithm, discussed in section 3.3.2, is based on the idea of building the Ansatz iteratively throughout the optimization process. In each iteration, we extend the Ansatz by a factor $e^{\theta_k \hat{A}_k}$, where $\hat{A}_k$ is chosen from a previously defined operator pool $\{\hat{A}_k\}$. To determine which operator to pick, we compute the energy gradient $\frac{\partial E}{\partial \theta}$ for every operator in the pool and choose the one with the largest gradient. After each iteration, we then perform a VQE routine to optimize the parameters. The key advantage of this approach is that it reduces the number of parameters and the circuit depth, compared to other Ansatz choices such as UCCSD. The main downside is the increase in the number of measurements, since we need to perform gradient measurements for each operator in the pool and a full VQE optimization in each iteration of the algorithm.
In order to translate this method to First Quantization, we consider the operator pool from the original ADAPT-VQE paper [25], consisting of the single- and double excitation operators

$$\hat{\tau}_k^a = a_a^\dagger a_k - a_k^\dagger a_a \qquad \text{and} \qquad \hat{\tau}_{kl}^{ab} = a_a^\dagger a_b^\dagger a_k a_l - a_k^\dagger a_l^\dagger a_a a_b \,. \tag{127}$$

The operator $a_a^\dagger a_k$ has the effect of removing an electron from the k-th orbital and adding an electron to the a-th orbital. If the k-th orbital is unoccupied or if the a-th orbital is occupied, this operator annihilates the state. In First Quantization, we represent the operators (127) using projection operators summed over all electron indices.

$$\hat{\tau}_k^a = \sum_{i=1}^{m} \Big(|\phi_a\rangle \langle \phi_k|_i - |\phi_k\rangle \langle \phi_a|_i \Big) \tag{128}$$

$$\hat{\tau}_{kl}^{ab} = \sum_{i,j=1}^{m} \Big(|\phi_a\rangle \langle \phi_k|_i |\phi_b\rangle \langle \phi_l|_j - |\phi_k\rangle \langle \phi_a|_i |\phi_l\rangle \langle \phi_b|_j \Big) \tag{129}$$

Under the first quantized fermion-to-qubit mapping (73) the operators $\hat{\tau}_k^a$ and $\hat{\tau}_{kl}^{ab}$ are mapped to a sum of Pauli strings. In order to estimate the number of Pauli strings in this sum, we observe that each term $|\phi_a\rangle \langle \phi_k|_i$ can be expressed as a tensor product of $n = \log N$ single qubit operators of the type $|i\rangle \langle j|$ for $i, j \in \{0, 1\}$. Each of these operators can be written as the sum of two Pauli matrices (including the identity), using the expressions in equation (76). The total number of Pauli strings in the two-electron operator (129) is thus equal to $\mathcal{O}(m^2 2^{2n}) = \mathcal{O}(m^2 2^{2 \log N}) = \mathcal{O}(m^2 N^2)$. At this point, it is important to note that this behaviour differs from the Second Quantization case. In fact, the second quantized two-electron operator in equation (127) can be expressed as a sum of 8 Pauli strings, regardless of the number of orbitals and electrons. Using a Trotter expansion we can approximate the operator $e^{\theta \hat{\tau}_{kl}^{ab}}$ as a product of individual Pauli strings

exponentials. In the following, we use P_λ^N to represent a Pauli string acting on N qubits. In First Quantization, each Pauli string appearing in the expansion can only act on a maximum of $2\log N$ qubits.

$$\text{Second Quantization:} \quad e^{\theta \hat{\tau}_{kl}^{ab}} \approx \prod_{\lambda=1}^{8} e^{i\theta P_\lambda^N} \tag{130}$$

$$\text{First Quantization:} \quad e^{\theta \hat{\tau}_{kl}^{ab}} \approx \prod_{\lambda=1}^{\mathcal{O}(m^2N^2)} e^{i\theta P_\lambda^{2\log N}} \tag{131}$$

In order to determine the necessary circuit depth in both cases, we need to consider the circuit implementation in terms of elementary quantum gates for the single Pauli string exponential $e^{i\theta P_\lambda}$.
Let $P^N = P_1 \otimes ... \otimes P_N$ be a Pauli string acting on N qubits, where $P_i \in \{X, Y, Z\}$. In order to implement $e^{i\theta P^N}$, we first observe that we can transform this operator into the exponential $e^{i\theta Z^N}$, where $Z^N = Z_1 \otimes ... \otimes Z_N$ is a string of Z operators [43].

$$e^{i\theta P^N} = \left(\bigotimes_{i=1}^{N} R_i^\dagger\right) e^{i\theta Z^N} \left(\bigotimes_{i=1}^{N} R_i\right) \tag{132}$$

The single qubit rotation gates R_i are chosen such that they satisfy $P_i^N = R_i^\dagger Z_i R_i$ and thus transform the two Pauli strings into each other, i.e. $P^N = (\bigotimes R_i^\dagger) Z^N (\bigotimes R_i)$. In particular we choose the following correspondence:

$$\begin{aligned} P_i^N = X_i \quad &\Rightarrow \quad R_i = H_i \\ P_i^N = Y_i \quad &\Rightarrow \quad R_i = R_x(\pi/2) \\ P_i^N = Z_i \quad &\Rightarrow \quad R_i = \mathbb{1}_i \end{aligned} \tag{133}$$

If $P_i^N = \mathbb{1}$, we can simply ignore the i-th qubit, as it is not affected by the operator $e^{i\theta P^N}$. It can be shown (see [43], p.210) that the Z-string exponential $e^{i\theta Z^N}$ can be implemented using a sequence of CNOT gates and a single qubit rotation around the z-axis by -2θ. Fig. 17 depicts the circuit implementing the four qubit exponential $e^{i\theta Z_1 Z_2 Z_3 Z_4}$.

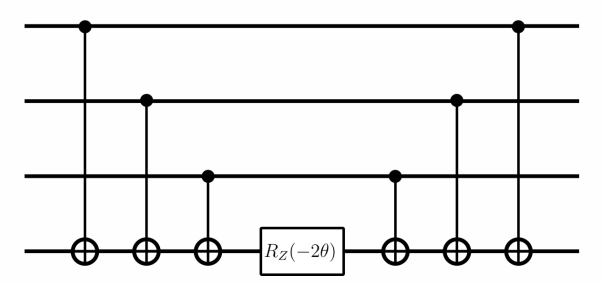


FIGURE 17: Circuit implementation of the operator $e^{i\theta Z^N}$ for $N = 4$.

The operator $e^{i\theta P^N}$ can therefore be implemented by simply adding a set of single qubit rotations R_i to the left of this circuit and set of single qubit rotations $R_i^\dagger$ to the right of it. As an example case, Fig. 18 depicts the circuit corresponding to the operator $e^{i\theta X_1 Y_3 Z_4}$.

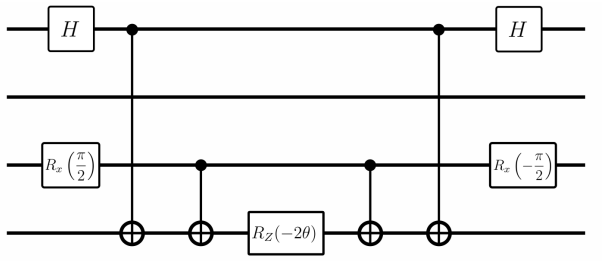


FIGURE 18: Circuit implementation of the operator $e^{i\theta X_1 Y_3 Z_4}$.

As a result, a circuit implementation of the Pauli string exponential $e^{i\theta P^N}$ requires $2(q-1)$ CNOT gates and $2q+1$ single qubit rotation gates, where q is the Pauli weight of P^N (the number of qubits that P^N acts on in a non-trivial way).

We now return to the original question of determining the circuit depth of the Ansatz operators (130) and (131). As we have seen, this circuit depth directly depends on the Pauli weight of those Pauli strings that appear when transforming the excitation operators $\hat{\tau}_k^a$ and $\hat{\tau}_{kl}^{ab}$ under a fermion-to-qubit mapping. In Second Quantization, the Jordan-Wigner and parity encoding, for example, result in a Pauli weight of $\mathcal{O}(N)$, implying a circuit depth of $\mathcal{O}(N)$ per Ansatz operator $e^{\theta\hat{\tau}_{ij}^{ab}}$. In fact, it is shown in [55] that the number of CNOT gates can be estimated as approximately $64N$. This result can be improved, however, when using a more efficient encoding such as the Bravyi-Kitaev transformation. As we have seen in section 3.2.3, this results in a Pauli weight of $\mathcal{O}(\log N)$ and thus a circuit depth of $\mathcal{O}(\log N)$ per Ansatz operator. This connection between the choice of encoding and the circuit depth of the Ansatz is crucial when deciding for an optimal VQE implementation.

In First Quantization, the encoding of the wavefunctions leads to $m \log N$ qubits. Since every Pauli string resulting from the transformation of $\hat{\tau}_k^a$ and $\hat{\tau}_{kl}^{ab}$ can only act on a maximum of two electrons, the maximum Pauli weight is equal to $2 \log N$. Each Pauli string exponential thus requires a circuit depth of $\mathcal{O}(\log N)$. However, since the Ansatz operator $e^{\theta\hat{\tau}_{ij}^{ab}}$ in equation (130) is given by a product of $\mathcal{O}(m^2N^2)$ of such Pauli string

exponentials, the circuit depth per operator would be $\mathcal{O}(m^2N^2 \log N)$. This represents a considerable disadvantage of First Quantization compared to Second Quantization when using an ADAPT-VQE based approach. Again, this disadvantage is a direct result of the fact that the number of Pauli strings in each excitation operator $\hat{\tau}^{ab}_{kl}$ is constant in Second Quantization but dependent on N in First Quantization.
The question now is whether we can choose an operator pool in First Quantization, such that each operator does not result in a higher number of Pauli strings compared to Second Quantization. It turns out that we have already encountered such an approach in the qubit-ADAPT-VQE Ansatz, discussed in section 3.3.2.

4.4.2 qubit-ADAPT-VQE

The qubit-ADAPT-VQE Ansatz differs from the original ADAPT-VQE method mainly in the definition of the operator pool. Instead of choosing the excitation operators $\hat{\tau}^a_k$ and $\hat{\tau}^{ab}_{kl}$, the pool now consists of those Pauli strings that appear in the transformed versions of these operators in qubit space. The Ansatz is then iteratively extended by operators of the form $e^{i\theta P^q}$, where P^q is a Pauli string acting on q qubits. As discussed in the previous section, in a Second Quantization approach, each Pauli string exponential requires a circuit depth of $\mathcal{O}(q)$, where $q = N$ for the Jordan-Wigner encoding and $q = \log N$ for the Bravyi-Kitaev encoding. Although numerical simulations have shown [55] that this approach reduces the overall gate depth compared to the original ADAPT-VQE, the asymptotic scaling of the circuit depth with N remains unchanged.
Using the Ansatz operators $e^{i\theta P^q}$ has the additional implication that in Second Quantization certain symmetries are not conserved. To see this, consider the operator $\hat{\tau}^1_3 = a^\dagger_1 a_3 - a^\dagger_3 a_1$. Under the Jordan-Wigner transformation, it is mapped to the qubit operator

$$\begin{aligned} \hat{\tau}^1_3 &= \sigma^+_1 Z_2 \sigma^-_3 - \sigma^-_1 Z_2 \sigma^+_3 \\ &= \frac{1}{2}\left(X_1 Z_2 Y_3 - Y_1 Z_2 X_3\right) \end{aligned} \tag{134}$$

Here, we have used the identity $\sigma^\pm_i = (X \mp iY)/2$. In qubit space, σ^+ and σ^- represent the addition and removal of electrons from fermionic modes. Since the operator $\hat{\tau}^1_3$ consists of an equal number of σ^+'s and σ^-'s and since Z does not affect the occupancy, the total number of particles is conserved. The individual Pauli strings in (134), however, do not preserve the particle number. The term $X_1 Z_2 Y_3$, for instance, would map the zero-electron state $|000\rangle$ to the 2-electron state $i\,|101\rangle$.
Using the qubit-ADAPT-VQE Ansatz in Second Quantization therefore does not guarantee that the number of particles stays constant throughout the optimization process. One could argue that it is a more heuristic Ansatz that works well in small-scale simulations but does not have rigorous performance guarantees.

We now turn to a first quantized version of the qubit-ADAPT-VQE method. Analogous to the Second Quantization approach, we choose the operator pool to consist of all Pauli strings that appear in the expansion of the excitation operators $\hat{\tau}_k^a$ and $\hat{\tau}_{kl}^{ab}$ in equations (128) and (129). More specifically, it is the set of all possible Pauli strings that act on either a single electron or a pair of electrons. Since each electron is represented by $n = \log N$ qubits and there are four possible Pauli operators (including the identity) for each qubit, the total number of operators in the pool is equal to $\mathcal{O}(m^2 4^{2\log N}) = \mathcal{O}(m^2 N^4)$. Furthermore, each Pauli string has a maximum Pauli weight of $\mathcal{O}(\log N)$ and thus each Ansatz operator $e^{i\theta P}$ requires a circuit depth of $\mathcal{O}(\log N)$.

In order to estimate the number of gradient measurements in each iteration, we need to multiply the number of operators in the pool with the number of Pauli strings that have to be measured to evaluate the commutator $[H, \hat{A}_i]$. Since H contains $\mathcal{O}(m^2 N^4)$ terms and $\hat{A}_i$ is a single Pauli string, we might expect the number of such Pauli strings to again be $\mathcal{O}(m^2 N^4)$. However, we need to keep in mind that $\hat{A}_i$ only acts on a maximum of two electrons. All terms in the Hamiltonian that do not have any overlap with these two electrons thus vanish in the commutator and we only need to consider those $\mathcal{O}(mN^4)$ terms that act on at least one of the two electrons. At this point, we might expect the number of gradient measurements to then be equal to #(operators in the pool)×(#non-vanishing terms in the commutator). However, this would result in measuring the same Pauli string more than once. To see this, we observe that any Pauli string in $[H, \hat{A}_i]$ that needs to be measured can only act on a maximum of three electrons (since the only non-vanishing terms are products of Pauli strings that overlap on either one or two electrons). We can therefore simply count the total number of Pauli strings acting on three electrons, which is equal to $\mathcal{O}(m^3 4^{3\log N}) = \mathcal{O}(m^3 N^6)$.

In order to compare these results to the case of Second Quantization, we remember that in the original version of qubit-ADAPT-VQE there are $\mathcal{O}(N^4)$ operators in the pool, $\mathcal{O}(N^4)$ terms in the Hamiltonian and thus $\mathcal{O}(N^8)$ gradient measurements are necessary in each iteration. Whether this scaling is better or worse than in First Quantization depends again on the molecular system we are interested in. For a small number of electrons, the scaling with respect to N might provide an advantage but for large m the scaling could be worse. We should note, that in both cases the number of measurements can be reduced by measuring commuting Pauli strings simultaneously. Next, in both First and Second the required circuit depth of each Ansatz operator shows a best-case scaling of $\mathcal{O}(\log N)$. In Second Quantization, this is only possible when using the Bravyi-Kitaev encoding, however. In contrast, the Jordan-Wigner or parity encoding results in a gate depth of $\mathcal{O}(N)$.

The key advantage of qubit-ADAPT-VQE (and, indeed, of any Ansatz) in First Quantization, is that it conserves the number of electrons. In fact, the fermion-to-qubit mapping simply cannot represent states that have more or fewer than m electrons. It is an interest-

ing observation that the choice of First or Second Quantization translates into choosing a physical property that is naturally conserved. Choosing First Quantization intrinsically preserves the number of particles but antisymmetrization must be enforced manually. In contrast, choosing Second Quantization intrinsically preserves antisymmetrization but the correct number of particles must be enforced manually. In the following, we explore how enforcing the antisymmetrization can be combined with the application of a variational Ansatz in First Quantization.

4.4.3 Antisymmetrization Issues

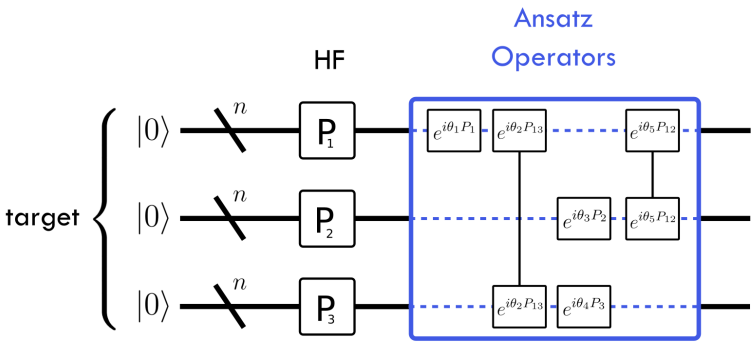
As discussed in section 4.2, implementing VQE in First Quantization requires us to apply an antisymmetrization circuit that maps any product state of single electron orbitals to its corresponding Slater determinant. This raises an important question in the context of the Ansatz choice. Should the antisymmetrization be applied before or after the parameterized Ansatz state has been prepared? In general, there are no fundamental reasons to prefer one over the other, but for specific Ansätze and specific antisymmetrization circuits there are some restrictions.

Consider first the case of applying the antisymmetrization before the variational Ansatz. This implies that the Ansatz operators must be chosen in such a way that they preserve the antisymmetry of the reference state. Since we have identified qubit-ADAPT-VQE as the most efficient candidate for an Ansatz in First Quantization, we have to determine whether operators of the form $e^{i\theta P}$ are antisymmetry preserving. Unfortunately, it can shown that this is not the case (see Appendix A.8).

We thus need to consider the second option of first applying the variational Ansatz operators and then performing the antisymmetrization. As we have seen in section 4.2, the antisymmetrization method proposed by Babbush et al in [13] exhibits an excellent scaling in the circuit depth of $\mathcal{O}(\log^c m \log\log N)$. However, this method has an important restriction in that it requires the input state to be sorted and repetition free. By "sorted" we mean that the any two electrons with indices i and j where $i > j$, must be in orbitals $|\phi_k\rangle_i$ and $|\phi_l\rangle_j$ such that $k > l$. By "repetition-free", we mean that no two electrons can occupy the same orbital. In the context of qubit-ADAPT-VQE this presents a problem, however. Even if the reference state $\left|\psi^{\mathrm{HF}}\right\rangle$ fulfills these criteria, the state $e^{i\theta P}\left|\psi^{\mathrm{HF}}\right\rangle$, in general, does not. To see this, consider again the example provided in Appendix A.8. However, it is possible solve this problem by slightly modifying the antisymmetrization circuit.

We begin by preparing the register `target` in a Hartree-Fock reference state. Then, a set of Ansatz operators of the type $e^{i\theta P}$ is applied, where P represents a Pauli string on one or two electrons. In the context of qubit-ADAPT-VQE, we iteratively build this sequence of operators and choose their type and location depending on the result of the gradient measurements. The following circuit depicts the case of $m = 3$ electrons and five Ansatz

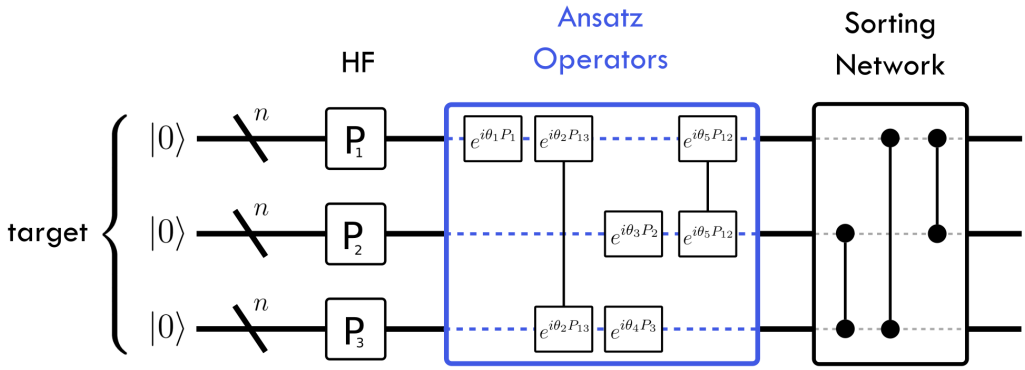
operators as an example.



In general, the state $|\psi\rangle$ resulting from this circuit is given by a superposition whose states are not necessarily sorted and may contain repetitions. In the following, we omit the normalization factor for clarity.

$$|\psi\rangle = \sum_{k_1 \ldots k_m} c_{k_1 \ldots k_m} |\phi_{k_1}\rangle \otimes \ldots \otimes |\phi_{k_m}\rangle \quad \text{where} \quad 0 \leq k_i \leq N\,. \tag{135}$$

The first step is sort every element in the superposition. This can be achieved by simply applying the same sorting network that we have used in the antisymmetrization circuit in section 4.2. The only difference is that we do not need to store the outcome of each comparator in an ancilla register.

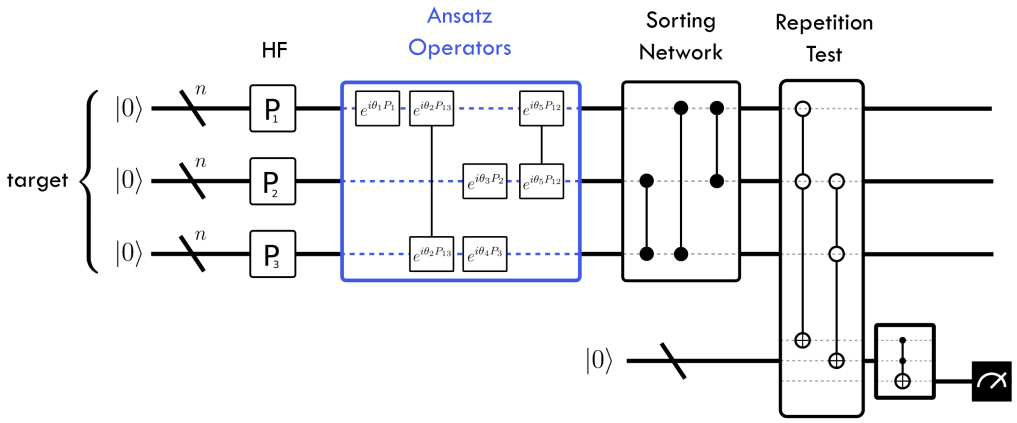


The resulting state is given by a superposition over states that are sorted but could still contain repetitions.

$$|\psi\rangle = \sum_{k_1 \geq \ldots \geq k_m} c_{k_1 \ldots k_m} |\phi_{k_1}\rangle \otimes \ldots \otimes |\phi_{k_m}\rangle \quad \text{where} \quad 0 \leq k_i < N\,. \tag{136}$$

Next, we need to project this state on the repetition-free subspace. This can again be achieved by repurposing the same repetition-check subroutine that we have already used in section (4.2). To summarize this step, we compare all adjacent pairs of electrons and

set an ancilla qubit to $|1\rangle$ if and only if the electrons are in different orbitals. If all ancilla qubits are in the state $|1\rangle$, then the electrons are in a repetition free state. By performing a measurement on the ancillas we can thus project the state of `target` onto the repetition-free subspace.



If the projection is successful, the final state of this circuit is a superposition of states that are both sorted and repetition-free.

$$|\psi\rangle = \sum_{k_1>\ldots>k_m} c_{k_1\ldots k_m} |\phi_{k_1}\rangle \otimes \ldots \otimes |\phi_{k_m}\rangle \quad \text{where} \quad 0 \leq k_i < N\,. \tag{137}$$

This state is now a valid input state that can be antisymmetrized using the method proposed by Babbush et al [13]. It is important to emphasize, however, that we have introduced a second repetition-check. The full preparation of the antisymmetrized Ansatz thus depends on the measurement outcome of two measurements and we only accept the result if both measurements are successful. Fig. 19 shows the entire circuit that integrates the original antisymmetrization circuit and the Ansatz operators.

Adding another instance of the sorting network and repetition-check to the antisymmetrization step increases the overall circuit depth. However, it is clear that it does not change the scaling of the circuit depth, circuit complexity, or number of ancilla qubits.

At this point we need to consider two questions. *i*) Is this Ansatz expressive enough? In other words, does the circuit in Fig. 19 produce states that can accurately approximate the real ground state? And *ii*) Is the probability of success large enough to be efficient in realistic situations?

Answering question *i*) is very difficult, if not impossible, if we do not have access to a numerical simulation with which to apply this method to a real system. Although there is a variety of publicly accessible code for simulations of VQE in Second Quantization, there is no such code in the case of First Quantization. Similarly, answering question *ii*) analytically is likely impossible. This is because the whole premise of adaptive Ansätze is

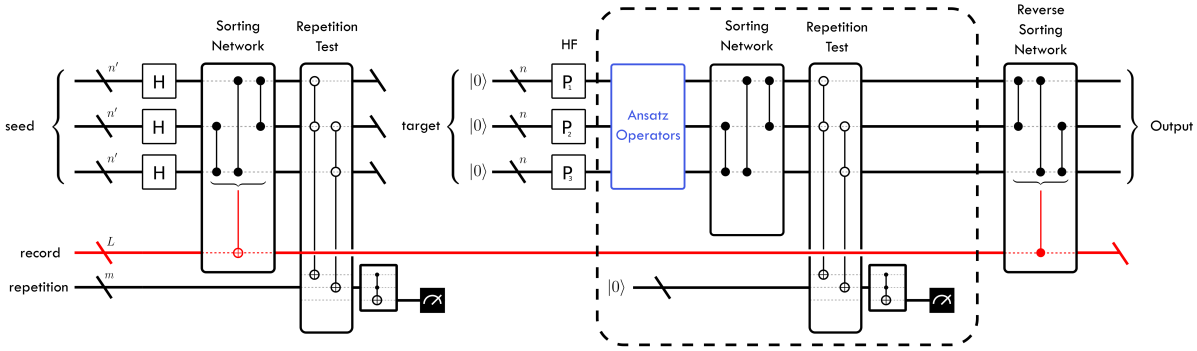


FIGURE 19: The circuit combining the variational Ansatz operators with the antisymmetrization method [13]. The circuit differs from Fig. 11 only in the dashed box. After preparing a HF reference state, variational Ansatz operators (blue box) are applied. In order to project the state onto the repetition-free subspace, a sorting network and repetition-check are performed. The resulting state can be antisymmetrized by applying the original sorting network in reverse.

that we do not know a priori which Ansatz operators are applied. Without this information, however, we do not know what state the electron registers are in and consequently cannot predict the probability of a successful projection onto the repetition-free subspace. In the following chapter, we take a first step towards answering these questions by implementing the circuit in Fig. 19 in Python. In particular, we focus on question *ii*) with the goal of determining the probability of success depending on the system size.

5 Numerical Simulations

A downside of most VQE Ansätze, especially of adaptive ones, is that there are not many rigorous performance guarantees. Consider the ADAPT-VQE Ansatz as an example. Although we expect the Ansatz state to approach the Full Configuration Interaction ground state in the limit of infinitely many Ansatz operators (see section 3.3.2), it is not clear how much the Ansatz deviates from this solution for a finite number of Ansatz operators. We thus rely heavily on classical simulations of VQE. Typically, the performance of a newly proposed Ansatz is evaluated by applying it to a small molecule using a small number of basis functions. The results are then compared to classical simulations to determine how well they approximate the real solution. Furthermore, we can compare different Ansätze in terms of the necessary circuit depth, circuit complexity or number of parameters.
Currently, there are several publicly accessible implementations of VQE in Second Quantization, such as `QISKIT.VQE` [48]. However, there are no such implementations for VQE in First Quantization yet. It is important to note that we cannot simply reuse existing code for this purpose. This is because the encoding is fundamentally different in First Quantization. We have to store the fermionic state in a different way, apply different Ansatz operators, and measure different Pauli strings that are combined to estimate the energy expectation value. The only step that is the same in First and Second Quantization is the classical optimization routine.
In this chapter, we take a first step towards implementing VQE in First Quantization. In particular, we use the Python package `QISKIT` to implement the circuit shown in Fig. 19, i.e. the antisymmetrization based on Babbush et al [13], combined with the application of Ansatz operators as used in the qubit-ADAPT-VQE Ansatz. We then use this simulation to estimate the probability of successfully generating Ansatz states, depending on the number of electrons m and number of basis functions N. The computational resources and required time connected to implementing the full qubit-ADAPT-VQE algorithm would go beyond the scope of this thesis. Therefore, instead of choosing the Ansatz operators based on gradient measurements of the energy of a real molecular system, we restrict ourselves to choosing the Ansatz operators at random. This, of course, is only an approximation to real situations, but it can serve as a first estimation of the probability of success and its behaviour as a function of the system size. Completing the VQE algorithm and applying it to a real system is a topic for a future project.
The code written for this project is available on GitHub [56].

5.1 QISKIT

`QISKIT` [48] is a Python based, open-source tool for working with and designing quantum circuits and quantum algorithms. On the lowest level, it allows us to build arbitrary quantum circuits by defining and applying individual quantum gates and performing measurements. These quantum programs can then be executed on real quantum devices on the IBM Quantum Experience [30] or on classically simulated devices. These simulators are able to replicate the noise and errors resulting from using real quantum devices. Any simulations in this thesis are performed using the IBM Aer Simulator.

5.2 Implementation of the Antisymmetrization Circuit

We begin with the implementation of the antisymmetrization circuit as proposed by Babbush et al [13], acting on a simple product state. We essentially follow the structure of the algorithm as described in section 4.2.3 and translate Step 1 to Step 4 into functions that can be executed in Python using the `QISKIT` package. For the purpose of clarity, we summarize each step here and explore what the resulting quantum circuit looks like using the example of $m = 2$ electrons and $N = 8$ basis functions. Fig. 20 depicts the entire circuit and Fig. 21 shows the same circuit in more detail.

Step 1: Prepare `seed`
In the first step of the algorithm, we initialize the registers `seed`, `record` and `repetition`. `seed` consists of m electrons, each given by $n = \log N$ qubits and labeled alphabetically ("a" for the first electron, "b" for the second on, etc.). In our example, we consider two electrons with 3 qubits each. In particular, we use the same number of qubits for each electron in `seed` and `target`, i.e. $n' = n$.
`seed` is initialized in the state $|0\rangle^{\otimes 6}$ and transformed by applying a Hadamard gate to each qubit.

Step 2: Sort `seed`
Next, we apply a sorting network to `seed`. Since in our example we only consider two electrons, the sorting network consists of a single comparator and `record` is given by a single qubit. Choosing $n' = 3$ implies that the reversible comparator requires two instances of `Compare-2`, requiring two additional ancilla qubits. After comparing the electrons a and b, we set the qubit in `record` using a controlled X-gate (control state '01') and uncompute `seed` by applying the reverse of the comparator. Finally, we perform a swap of the two electrons if `record` is in the state $|1\rangle$. As $n' = 3$, this step requires three instances of CSWAP.

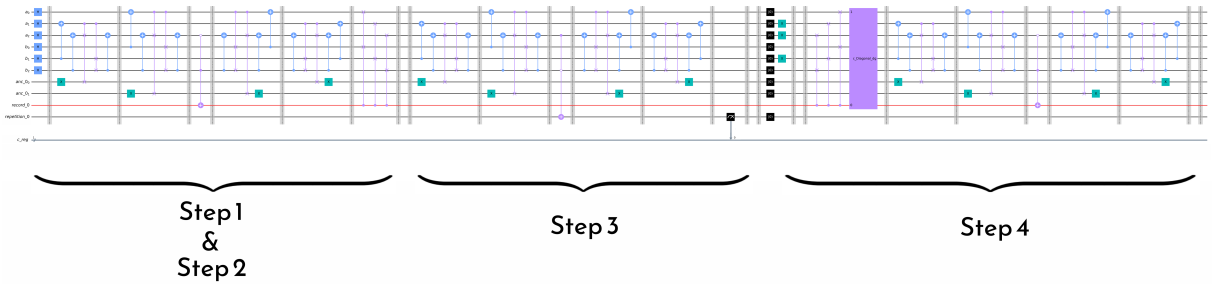


FIGURE 20: The antisymmetrization circuit based on [13] for $m = 2$ electrons and $N = 8$ basis functions. Step 1: prepare seed in a superposition of all states; Step 2: sort seed using a reversible sorting network; Step 3: perform a repetition-check and measurement; Step 4: Initialize the electrons in product state and apply the reverse sorting network including a controlled phase gate.

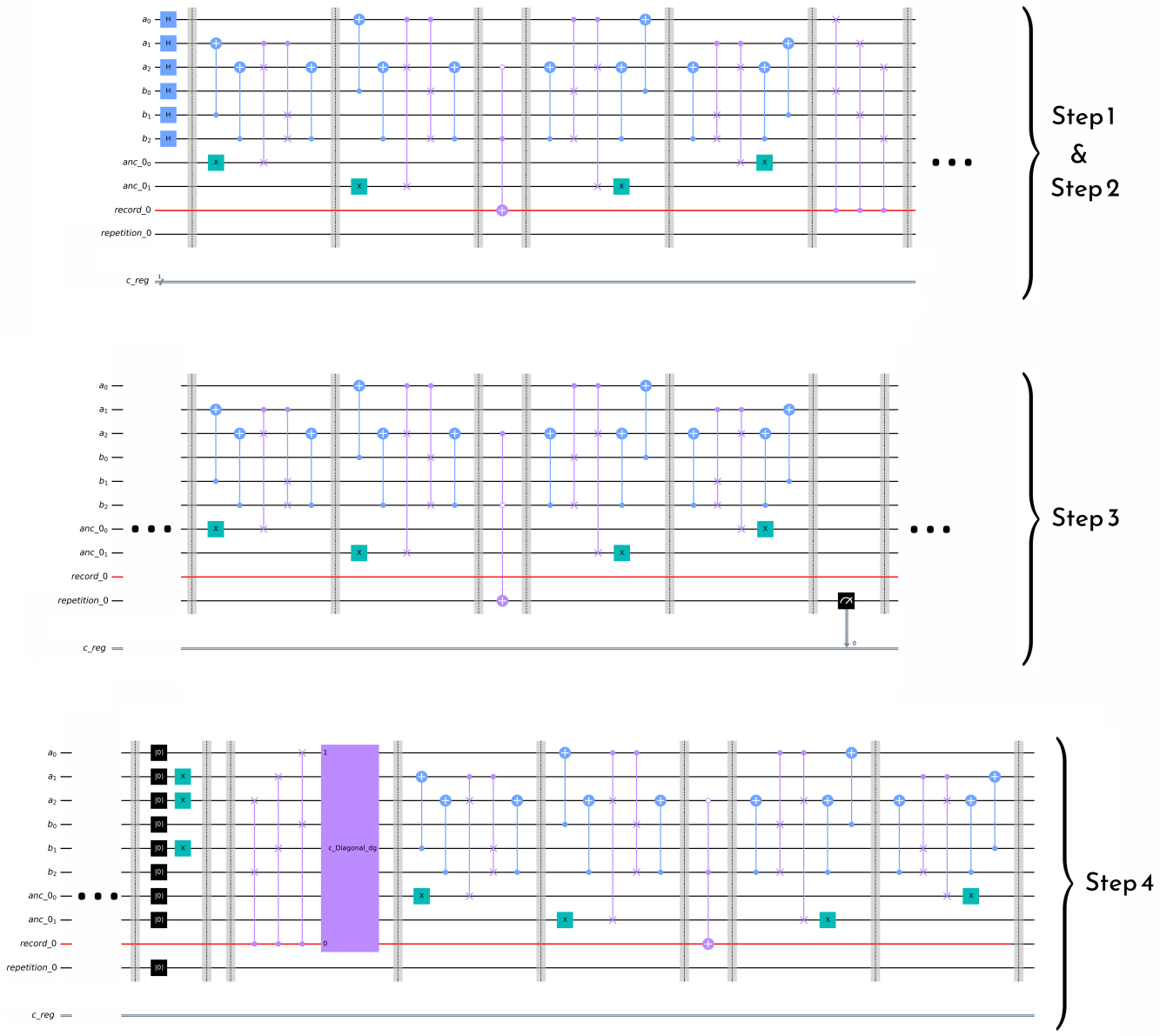


FIGURE 21: Zooming into the steps of the circuit shown in Fig. 20.

Step 3: Delete repetitions from `seed`
In the case of $m = 2$, the repetition-check requires the application of only one comparator and thus only one ancilla qubit in the register `repetition`. Analogous to Step 2, we apply two instances of `Compare-2`, such that qubits a_2 and b_2 have the same relation towards each other as a and b. The only difference, however, is that the controlled X-gate now has a control state of '10' (as opposed to '01' in Step 2). After uncomputing `seed`, we measure the register `repetition` and store the outcome in a classical register `c_reg`. If the outcome of the measurement is $|1\rangle$, the preparation of `record` is successful and it is now disentangled from `seed`.

Step 4: Apply the reverse of the sorting network to `target`
At this point, we can discard `repetition` and re-initialize all other qubits to $|0\rangle$. Since we have chosen $n' = n$, we can simply reuse `seed` and call it `target`. We initialize `target` to an arbitrary product state by applying single qubit Pauli rotations. In our example we choose the state $|011\rangle\,|010\rangle = |\phi_3\rangle\,|\phi_2\rangle$, achieved by three X-gates. Next, we apply the reverse of the sorting network to `target` to carry out its antisymmetrization. In practice, this corresponds to applying all gates that were used in Step 2, but in reverse order. To distinguish antisymmetrization from symmetrization we include an additional cPHASE gate, which applies a factor of -1 to the qubit a_0 if `record` is in the state $|1\rangle$. Intuitively, applying the comparator in reverse "erases" the state of `record` while keeping `target` in a superposition of different orderings. Finally, we can discard all ancilla qubits and keep only the register `target` as output.

Note that the circuits in Fig. 20 and 21 contain barriers (grey, vertical lines) that are included for visual clarity. These barriers do not change the unitary described by the circuit. Moreover, if we delete these barriers, the circuit depth is reduced significantly, as `QISKIT` automatically performs circuit optimizations. For example, many gates can be applied simultaneously and some gates even cancel each other. An example of the same circuit without barriers is provided in Appendix B.1.

5.3 Implementation of the Ansatz Operators

We now turn to the implementation of the Ansatz operators used in the First Quantization version of qubit-ADAPT-VQE. In particular, these are operators of the type $e^{i\theta P}$, where θ is a variational parameter and P is a Pauli string acting on either one or two electrons. As already described in section 4.4.1, such Pauli string exponentials can be translated into elementary quantum gates using a simple ladder structure of CNOT gates and two sets of single qubit rotations. To illustrate this, Fig. 22 shows an example of three such Ansatz operators acting on a register of $m = 2$ electrons, each given by $n = 3$ qubits.

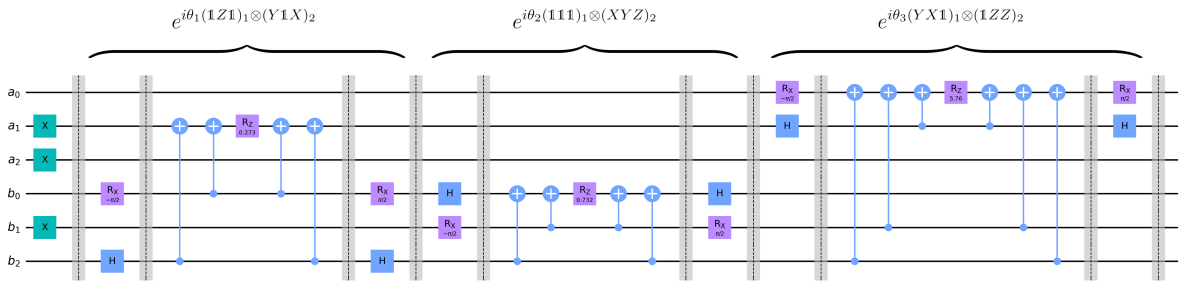


FIGURE 22: Example of three Ansatz operators in the first quantized version of qubit-ADAPT-VQE. Each Pauli string exponential is implemented using a ladder structure of CNOTs on all qubits that are acted on non-trivially by the Pauli string. Two sets of single qubit rotations transform the given Pauli string into the corresponding Z-string.

Finally, we need to combine these variational Ansatz operators with the antisymmetrization circuit. As established in section 4.4.3, this is achieved by applying a sorting network and repetition-check immediately following the Ansatz operators (see Fig. 19). The full circuit for the case of $m = 2$ and $N = 8$ is depicted in Appendix B.2.

5.4 Results

At this stage, we are able to apply arbitrary Ansatz operators and generate an antisymmetrized Ansatz state. As mentioned in the previous section, the success of this procedure depends on two measurement outcomes. The first measurement is necessary for the correct preparation of `record` and in the following we denote by $P_{\texttt{record}}$ the probability that this measurement is successful, i.e. equal to $'1'$. The second measurement deletes all repetitions from `target` following the application of the Ansatz operators and we denote the corresponding probability of success by $P_{\texttt{target}}$. Since we do not have access to a full adaptive VQE implementation, we restrict ourselves to choosing the Ansatz operators $e^{i\theta P}$ at random. In particular, we denote the number of Ansatz operators by M and randomly choose the parameter θ, the Pauli string P, and the electron(s) that the Pauli string acts on. To get an idea of how the probability of success changes as a function of the number of electrons, orbitals, and Ansatz depth, we perform the full state preparation for several different values of m, n, and M. Since the Pauli strings are selected at random, the resulting circuits can vary significantly. In order to reduce this variation, for each set of values n, m, and M, we generate 20 different instances of Ansatz operators and average the result. Finally, each circuit is executed 1000 times using Aer Simulator by specifying `qiskit.shots=1000`. The probabilities $P_{\texttt{record}}$ and $P_{\texttt{target}}$ are computed by counting the fraction of successful measurement outcomes. The total probability of success $P_{\texttt{total}} = P_{\texttt{record}}P_{\texttt{target}}$ is obtained by multiplying both probabilities.
In order to study the dependence on the depth of the Ansatz, figure 23 shows the results for $m = 2$ electrons and $n = 2, 3, 4$ qubits per electron, i.e. $N = 4, 8, 16$ basis functions.

The number of Ansatz operators is chosen in the range from 0 to 30. Since $P_{\texttt{record}}$ does not depend on M, we depict $P_{\texttt{target}}$ as a solid line and $P_{\texttt{total}}$ as a dashed line. Figure 24 shows the results for $m = 3$ electrons and $n = 2, 3$ qubits per electron.

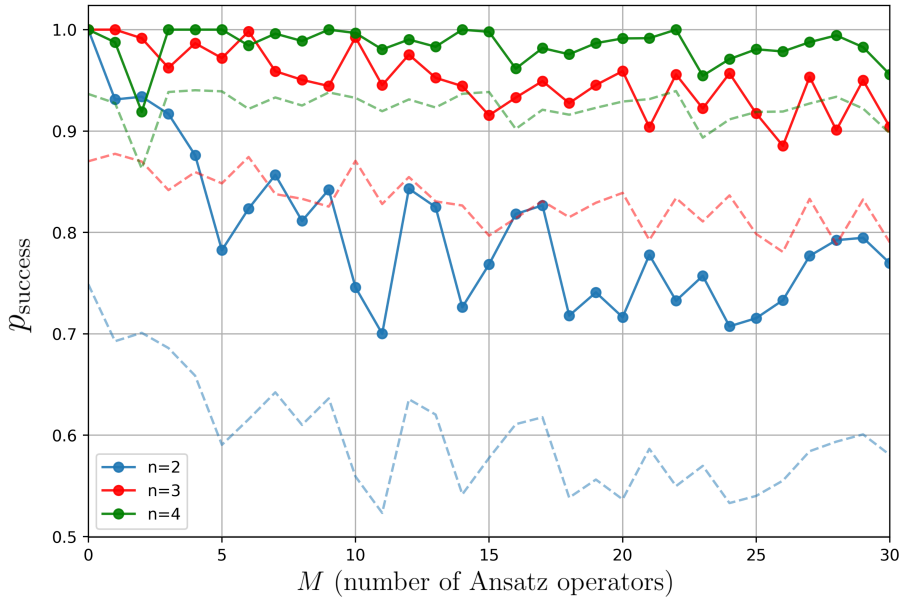


FIGURE 23: The probability of success for $m = 2$ electrons. n is the number of qubits per electron and $N = 2^n$ is the number of single electron basis functions. The solid line depicts $P_{\texttt{target}}$ and the dashed line depicts $P_{\texttt{total}} = P_{\texttt{target}} P_{\texttt{record}}$. The Ansatz operators are chosen randomly from a uniform distribution. Each data point is averaged over 20 different instances and each circuit is executed with 1000 shots.

FIGURE 24: The probability of success for $m = 3$ electrons. n is the number of qubits per electron and $N = 2^n$ is the number of single electron basis functions. The solid line depicts $P_{\texttt{target}}$ and the dashed line depicts $P_{\texttt{total}} = P_{\texttt{target}} P_{\texttt{record}}$. The Ansatz operators are chosen randomly from a uniform distribution. Each data point is averaged over 20 different instances and each circuit is executed with 1000 shots.

Our first observation within these results is that the probability of success gradually decreases with the number of Ansatz operators. This is to be expected since a larger Ansatz implies that the superposition contains a higher number of elements with repetitions. However, the probability of success increases globally with the number of basis functions N. By choosing $m = 2$ and $N = 16$, for example, $P_{\texttt{target}}$ can be kept above 0.9 for $M \leq 30$. It is important to note that, in practice, although it is beneficial to keep the probability of success as large as possible, it is not necessary to choose $P_{\texttt{total}}$ close to 1. This is because even if $P_{\texttt{total}} = 0.5$, we are still able to prepare a valid Ansatz after only two tries on average. Note that for $m = 2$ electrons, the probability of success is larger than 0.5 for all values of n.

To study the dependence on the number of basis functions further, figures 25 and 26 show the probability of success as a function of the number of qubits per electron n. Note that the number of basis functions is given by $N = 2^n$. In this case, we choose a fixed number of $M = 50$ Ansatz operators for $m = 2$, and $M = 20$ Ansatz operators for $m = 3$. Again, we observe that both $P_{\texttt{record}}$ and $P_{\texttt{target}}$ approach a probability of 1 with increasing n. These results are compatible with the hypothesis that First Quantization VQE may be particularly efficient when describing a small number of electrons with a large number of basis functions. In this case, the probability of successfully preparing an antisymmetrized Ansatz state naturally approaches 1. On the other hand, this method of preparing Ansatz states may be relatively inefficient if we are interested in cases where m and N are of similar size. In that case, the probability of success may decrease significantly such that each state preparation is only successful after a large number of repetitions.

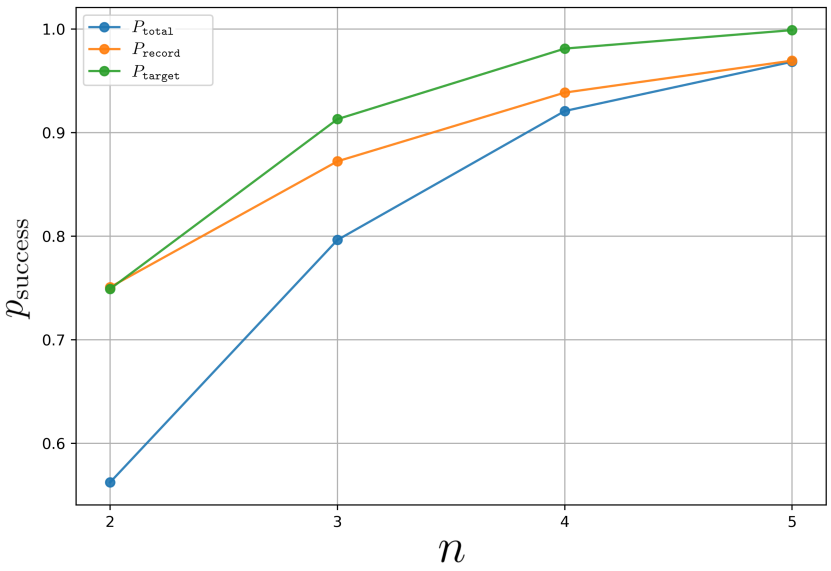


FIGURE 25: Probability of success for $m = 2$ electrons as a function of the number of qubits n per electron (or $N = 2^n$ basis functions). Each circuit contains $M = 50$ randomly chosen Ansatz operators, each data point is averaged over 50 random instances, and each circuit is executed 1000 times.

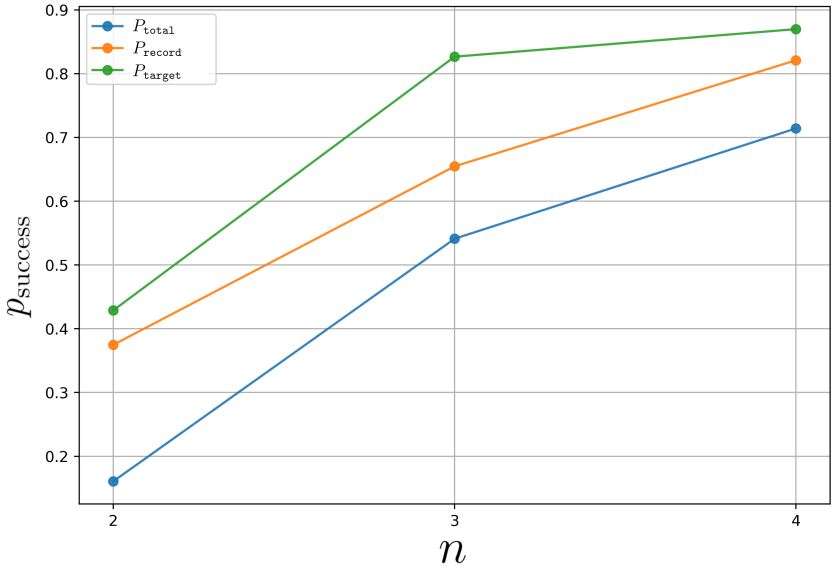


FIGURE 26: Probability of success for $m = 3$ electrons as a function of the number of qubits n per electron (or $N = 2^n$ basis functions). Each circuit contains $M = 20$ randomly chosen Ansatz operators, each data point is averaged over 20 random instances, and each circuit is executed 1000 times.

6 Summary and Outlook

The goal of this thesis was to explore possible approaches to implementing the Variational Quantum Eigensolver in a First Quantization framework. In particular, we asked the following questions: *i*) can every step of the algorithm be implemented efficiently in First Quantization? and *ii*) what are the advantages and disadvantages in performance compared to Second Quantization approaches? To answer these questions, we examined the main components of the VQE algorithm separately, translated them from Second Quantization to First Quantization and compared the performance in both cases in terms of the required number of qubits, the circuit depth, and the number of measurements.

The first step in VQE is to perform a fermion-to-qubit mapping. This transformation takes a fermionic Fock state and encodes it in the state of qubits in such a way that the fermionic nature of the state is preserved. In Second Quantization, this is achieved by transforming the creation and annihilation operators to corresponding qubit operators that satisfy the same anti-commutation relations. When describing m electrons in N fermionic modes, these encodings thus require N qubits. In First Quantization, however, we encode the wavefunction directly using a "binary" encoding, which requires $m \log N$ qubits. Especially in cases where we are interested in a small number of electrons and a large number of basis functions, this may reduce the number of qubits significantly. This difference is a natural consequence of the intrinsic differences between First and Second Quantization. In Second Quantization, we work with the orbitals and their occupancy. This makes it natural to represent states of indistinguishable particles, and in the context of VQE, antisymmetry is automatically preserved. The drawback that comes with this framework is its ability to describe states of 0 to N electrons. Since we are only interested in a constant number of m electrons, we therefore necessarily work in a Hilbert space that is "too large". Moreover, although physical errors in the experiment cannot change the antisymmetry of the encoded state, they can affect the number of particles it describes. In First Quantization, on the other hand, we work with the wavefunction of m particles directly. This introduces a physical requirement for a constant number of particles. However, we now have to manually enforce antisymmetry using an additional quantum circuit that is not necessary in Second Quantization. Whereas physical errors cannot change the number of particles, they may lead to a loss of antisymmetry. When deciding whether to use First or Second Quantization, we thus have to consider this trade-off and decide whether we prioritize the conservation of the number of particles or the antisymmetry of the state. Studying this problem numerically would be a topic for possible future research. Specifically, one could perform simulations of VQE in First and Second Quantization and investigate the effects of physical errors, for example using `QISKIT` simulators that replicate the noise in real quantum computers.

The second step in VQE is to transform the molecular Hamiltonian using a fermion-to-

qubit encoding and expand it as a sum of Pauli strings. The number of Pauli strings can be shown to be $\mathcal{O}(N^4)$ in Second Quantization and $\mathcal{O}(m^2N^4)$ in First Quantization. Estimating the energy expectation value in each iteration of VQE thus requires a higher number of measurements in First Quantization. However, in both First and Second Quantization, several optimization strategies can be applied to reduce the number of measurement. As an example, we may simultaneously measure commuting groups Pauli strings.
The third step is the application of an antisymmetrization circuit. As this step is not necessary in Second Quantization, it leads to an additional circuit depth in any First Quantization approach. However, Babbush et al [13] proposed a method to carry out the antisymmetrization in a very efficient way, leading to a gate depth of $\mathcal{O}(\log^c m \log\log N)$ and an ancilla register of size $\mathcal{O}(m \log N)$. This step can thus be performed without contributing to the overall resource scaling. It is important to note that the antisymmetrization is not deterministic. In particular, it is necessary to perform a measurement whose outcome determines whether the state preparation was successful. The probability of success can be set above an arbitrary threshold by increasing the number of qubits per electron. Furthermore, the successful antisymmetrization requires the input state to be ordered and repetition free. We address this problem in the context of Ansatz choices.
Next, we explored a specific measurement strategy based on plane waves. We followed previous work by Babbush et al. [8], who introduced this framework in the context of Second Quantization and general quantum chemistry simulations. The key point of this method is that the kinetic part T of the molecular Hamiltonian is diagonal in the plane wave basis, making it very efficient to estimate its expectation value for a given state. By applying a discrete Fourier transform, we then perform a basis change into the plane wave dual basis. In this basis, the potential part $U + V$ of the Hamiltonian is diagonal and can thus be sampled efficiently. The expectation value of the Hamiltonian can then be estimated by measuring T in the plane wave basis, applying the circuit implementing a discrete Fourier transform, and measuring $U + V$. It can be shown that the number of measurements necessary to estimate $\langle H \rangle$ to an accuracy of ε scales as $\mathcal{O}(\varepsilon^{-2}(m^{2/3}N^{4/3} + m^{10/3}N^{2/3}))$. The scaling with the number of orbitals N has thus decreased significantly from $\mathcal{O}(m^2N^4)$. It is particularly efficient when describing molecules with a small number of electrons using a large number of basis functions. The disadvantage of using this method in Second Quantization is that applying a discrete Fourier transform requires a circuit depth $\mathcal{O}(N^2)$ (or $\mathcal{O}(N \log N)$ when using a Fast Fourier transform). First Quantization has the advantage that this step only requires a gate depth of $\mathcal{O}(\log^2 N)$. The disadvantage of First Quantization, however, is that the preparation of a reference state is not straightforward. Typically, this reference state is a Hartree-Fock state resulting from a classical calculation using localized (e.g. Gaussian) orbitals. Such a reference state can be applied by a set of single qubit rotations of constant depth. Preparing a reference state in the plane wave basis, however, would require us to either perform a classical calculation in the plane wave

basis or perform a basis change, which may result in a large circuit depth.

Next, we explored different Ansatz choices that are typically used in Second Quantization. We came to the conclusion that the Hardware-Efficient Ansatz, the Unitary Coupled Cluster Ansatz, and the ADAPT-VQE Ansatz exhibit significantly larger circuit depths in First Quantization. The main reason for this is that excitation operators of the form $a_i^\dagger a_j$ can be expanded as a constant number of Pauli strings in Second Quantization but require $\mathcal{O}(N)$ Pauli strings in First Quantization. Operators of the type $e^{\theta a_i^\dagger a_j}$ thus result in a larger gate depth. We identified qubit-ADAPT-VQE as an Ansatz that does not suffer from this problem. In qubit-ADAPT-VQE, we choose an operator pool consisting of all Pauli strings resulting from the excitation operators $\hat{\tau}_i^a$ and $\hat{\tau}_{ij}^{ab}$. In each iteration of the algorithm we choose an operator from the pool and extend the Ansatz using the factor $e^{i\theta P}$. In Second Quantization, the operator $e^{i\theta P}$ requires a circuit depth of $\mathcal{O}(N)$ when using the Jordan-Wigner encoding and $\mathcal{O}(\log N)$ when using the Bravyi-Kitaev encoding. In First Quantization, it requires a circuit depth of $\mathcal{O}(\log N)$, thus making it an efficient alternative. Furthermore, the number of gradient measurements in each iteration scales as $\mathcal{O}(m^3 N^6)$, compared to $\mathcal{O}(N^8)$ in Second Quantization. Whether this is an improvement depends on the situation we are interested in. If we describe a small number of electrons using a large number of basis functions, First Quantization may provide an advantage.

Finally, we investigated the problem of integrating the application of arbitrary variational Ansatz operators and the antisymmetrization of the Ansatz state. In general, there are two possibilities. We could either apply the antisymmetrization circuit directly to the reference state and implement the Ansatz operators afterwards or we apply the Ansatz operators to the reference state and perform the antisymmetrization afterwards. The former has the problem that arbitrary Ansatz operators generally do not preserve the antisymmetry. The latter has the problem that the antisymmetrization requires its input state to be ordered and repetition-free, properties that are not guaranteed for any Ansatz operator, in particular for the ones used in qubit-ADAPT-VQE. To solve this problem, we propose an adjustment to the original antisymmetrization circuit. Namely, we apply another instance of both the sorting network and the repetition-check, after the Ansatz was applied to the reference state. This method enables us to use any type of Ansatz operator and transform the resulting state into a valid input of the antisymmetrization circuit. Although the overall resource scaling of the antisymmetrization does not change, we have introduced a second measurement that determines whether the state preparation was successful. In the context of VQE, we only accept the prepared Ansatz state if both measurements are successful.

To estimate the behaviour of the probability of success, we perform numerical simulations. In particular, we implement the combined circuit for the application of Ansatz operators and the antisymmetrization in Python using `QISKIT`. Due to time and resource constraints, we do not implement the full adaptive VQE routine and thus restrict ourselves

to using a random sequence of Ansatz operators. We examine the probability of success as a function of the number of electrons m, the number of basis functions N, and the number of operators in the Ansatz M. We observe that for a fixed m, the probability of success quickly approaches 1 if N is increased. Moreover, the probability of success only decreases gradually with the number of Ansatz operators. This provides a promising outlook for the first quantized version of qubit-ADAPT-VQE to be an efficient candidate of VQE in First Quantization. Evidently, in order to demonstrate whether this method provides an advantage over Second Quantization approaches in real situations, we need access to a full numerical simulation of the VQE algorithm in First Quantization. We have taken a first step in this direction by implementing the necessary quantum circuits, but the measurement and optimization procedure are topics for a future research project.

References

[1] Aaronson, Scott: *Quantum Computing Since Democritus*. ISBN: 9780521199568, 2013

[2] Abrams, Daniel S. ; Lloyd, Seth: Simulation of Many-Body Fermi Systems on a Universal Quantum Computer. In: *Phys. Rev. Lett.* 79 (1997), Sep, 2586–2589. http://dx.doi.org/10.1103/PhysRevLett.79.2586. – DOI 10.1103/PhysRevLett.79.2586

[3] Aharonov, Dorit: *A Simple Proof that Toffoli and Hadamard are Quantum Universal*. arXiv:quant-ph/0301040, 2009

[4] Aharonov, Dorit ; Ben-Or, Michael: *Fault-Tolerant Quantum Computation with Constant Error Rate*. SIAM Journal on Computing. 38 (4): 1207–1282. doi:10.1137/S0097539799359385, 2008

[5] Ajagekar, Akshay ; You, Fengqi: Quantum computing and quantum artificial intelligence for renewable and sustainable energy: A emerging prospect towards climate neutrality. In: *Renewable and Sustainable Energy Reviews* 165 (2022), 112493. http://dx.doi.org/https://doi.org/10.1016/j.rser.2022.112493. – DOI https://doi.org/10.1016/j.rser.2022.112493. – ISSN 1364–0321

[6] Ajtai, M. ; Komlós, J. ; Szemerédi, E.: *Proceed- ings of the Fifteenth Annual ACM Symposium on Theory of Computing*. STOC '83 (ACM, New York, NY, USA, pp. 1-9, 1983

[7] *Altius*. https://www.altius.com/en/news/quantum-computing-the-next-major-technological-revolution-is-knocking-at-the-door/, 2022. – Accessed: 2023-01-27

[8] Babbush, Ryan ; Wiebe, Nathan ; McClean, Jarrod ; McClain, James ; Neven, Hartmut ; Chan, Garnet Kin-Lic: Low-Depth Quantum Simulation of Materials. In: *Phys. Rev. X* 8 (2018), Mar, 011044. http://dx.doi.org/10.1103/PhysRevX.8.011044. – DOI 10.1103/PhysRevX.8.011044

[9] Batcher, K. E.: *Sorting networks and their applications*. Spring Joint Computer Conference, Goodyear Aerospace Corporation, Akron, Ohio, 1968

[10] Bauer, Bela ; Bravyi, Sergey ; Motta, Mario ; Chan, Garnet Kin-Lic: Quantum Algorithms for Quantum Chemistry and Quantum Materials Science. In: *Chemical Reviews* 120 (2020), oct, Nr. 22,

12685–12717. http://dx.doi.org/10.1021/acs.chemrev.9b00829. – DOI 10.1021/acs.chemrev.9b00829

[11] Bellare, Mihir ; Kilian, Joe ; Rogaway, Phillip: The Security of the Cipher Block Chaining Message Authentication Code. In: *Journal of Computer and System Sciences* 61 (2000), Nr. 3, 362-399. http://dx.doi.org/https://doi.org/10.1006/jcss.1999.1694. – DOI https://doi.org/10.1006/jcss.1999.1694. – ISSN 0022–0000

[12] Bennett, C. H. ; Brassard, G.: *Quantum cryptography: Public key distribution and coin tossing.* https://doi.org/10.48550/arXiv.2003.06557, 1984

[13] Berry, Dominic W. ; Kieferov, Maria ; Scherer, Artur ; Sanders, Yuval R. ; Low, Guang H. ; Wiebe, Nathan ; Gidney, Craig ; Babbush, Ryan: Improved techniques for preparing eigenstates of fermionic Hamiltonians. In: *npj Quantum Inf* 4 (2018). http://dx.doi.org/https://doi.org/10.1038/s41534-018-0071-5. – DOI https://doi.org/10.1038/s41534–018–0071–5

[14] Bharti, Kishor ; Cervera-Lierta, Alba ; Kyaw, Thi H. ; Haug, Tobias ; Alperin-Lea, Sumner ; Anand, Abhinav ; Degroote, Matthias ; Heimonen, Hermanni ; Kottmann, Jakob S. ; Menke, Tim ; Mok, Wai-Keong ; Sim, Sukin ; Kwek, Leong-Chuan ; Aspuru-Guzik, Alán: *Noisy intermediate-scale quantum (NISQ) algorithms.* American Physical Society Rev. Mod. Phys. Vol 94 No 1 10.1103/RevModPhys.94.015004, 2022

[15] Blunt, Nick S. ; Camps, Joan ; Crawford, Ophelia ; Izsák, Róbert ; Leontica, Sebastian ; Mirani, Arjun ; Moylett, Alexandra E. ; Scivier, Sam A. ; Sünderhauf, Christoph ; Schopf, Patrick ; Taylor, Jacob M. ; Holzmann, Nicole: *Perspective on the Current State-of-the-Art of Quantum Computing for Drug Discovery Applications*

[16] Bova, Francesco ; Goldfarb, Avi ; Melko, Roger G.: Commercial applications of quantum computing. In: *EPJ Quantum Technology* 8 (2021). http://dx.doi.org/https://doi.org/10.1140/epjqt/s40507-021-00091-. – DOI https://doi.org/10.1140/epjqt/s40507–021–00091–

[17] Bravyi, S. ; Kitaev, A.: *Fermionic quantum computation.* Annals of Physics, 298, 210 quant-ph/0003137v2, 2002

[18] Cruz, P. M. Q. ; Catarina, G. ; Gautier, R. ; Fernández-Rossier, J.: *Optimizing quantum phase estimation for the simulation of Hamiltonian eigenstates.* Quantum Sci. Technol. 5 044005 https://doi.org/10.1088/2058-9565/abaa2c, 2020

[19] Echenique, Pablo ; Alonso, J. L.: A mathematical and computational review of Hartree-Fock SCF methods in quantum chemistry. In: *Molecular Physics* 105 (2007), dec, Nr. 23-24, 3057–3098. http://dx.doi.org/10.1080/00268970701757875. – DOI 10.1080/00268970701757875

[20] Einstein, A. ; Podolsky, B. ; Rosen, N.: Can Quantum-Mechanical Description of Physical Reality Be Considered Complete? In: *Phys. Rev.* 47 (1935), May, 777–780. http://dx.doi.org/10.1103/PhysRev.47.777. – DOI 10.1103/PhysRev.47.777

[21] Feynman, Richard: *Simulating physics with computers*. https://people.eecs.berkeley.edu/ christos/classics/Feynman.pdf. Version: 1982

[22] Ghirardi, GianCarlo: Entanglement, Nonlocality, Superluminal Signaling and Cloning. Version: apr 2013. http://dx.doi.org/10.5772/56429. In: *Advances in Quantum Mechanics*. InTech, apr 2013. – DOI 10.5772/56429

[23] Gidney, Craig ; Ekerå, Martin: *How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits*. https://doi.org/10.22331/q-2021-04-15-433, Quantum 5, 433, 2021

[24] Grimsley, Harper R. ; Claudino, Daniel ; Economou, Sophia E. ; Barnes, Edwin ; Mayhall, Nicholas J.: Is the Trotterized UCCSD Ansatz Chemically Well-Defined? In: *Journal of Chemical Theory and Computation* 16 (2020), Nr. 1, 1-6. http://dx.doi.org/10.1021/acs.jctc.9b01083. – DOI 10.1021/acs.jctc.9b01083. – PMID: 31841333

[25] Grimsley, Harper R. ; Economou, Sophia E. ; Barnes, Edwin ; Mayhall, Nicholas J.: *An adaptive variational algorithm for exact molecular simulations on a quantum computer*. Nature Communications volume 10, Article number: 3007. https://doi.org/10.1038/s41467-019-10988-2, 2019

[26] Hatano, Naomichi ; Suzuki, Masuo: Finding Exponential Product Formulas of Higher Orders. Version: nov 2005. http://dx.doi.org/10.1007/11526216_2. In: *Quantum Annealing and Other Optimization Methods*. Springer Berlin Heidelberg, nov 2005. – DOI 10.1007/11526216_2, 37–68

[27] Hohenberg, P. ; Kohn, W.: *Inhomogeneous Electron Gas*. 1, Physical Review 136, B864, 1964

[28] *IBM Osprey*. https://www.ibm.com/quantum/systems, 2022. – Accessed: 2023-01-27

[29] *IBM Quantum Computing Roadmap.* `https://www.ibm.com/quantum/roadmap`, 2022. – Accessed: 2023-01-27

[30] *IBM Q Experience.* IBM Quantum. https://quantum-computing.ibm.com/, 2021, 2023

[31] Jensen, F.: *Introduction to computational chemistry.* Chichester, UK Hoboken, NJ: Wiley; . ISBN 978-1-118-82595-2., 2017

[32] Jiang, Zhang ; Kalev, Amir ; Mruczkiewicz, Wojciech ; Neven, Hartmut: Optimal fermion-to-qubit mapping via ternary trees with applications to reduced quantum states learning. In: *Quantum* 4 (2020), Juni, 276. `http://dx.doi.org/10.22331/q-2020-06-04-276`. – DOI 10.22331/q–2020–06–04–276. – ISSN 2521–327X

[33] Jordan, P. ; Wigner, E.: *Über das Paulische Äquivalenzverbot.* Zeitschrift für Physik, Volume 47, Issue 9-10, pp. 631-651, doi= 10.1007/BF01331938, 1928

[34] Kandala, Abhinav ; Mezzacapo, Antonio ; Temme, Kristan ; Takita, Maika ; Chow, Markus Brinkand Jerry M. ; Gambetta, Jay M.: *Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets.* Nature 549, 242–246 https://doi.org/10.1038/nature23879, 2017

[35] Kitaev, A. Y.: *Fault-tolerant quantum computation by anyons.* Annals of Physics. 303 (1): 2–30. arXiv:quant-ph/9707021. doi:10.1016/S0003-4916(02)00018-0, 2003

[36] Kitaev, Alexei Y.: Quantum measurements and the Abelian Stabilizer Problem. In: *Electron. Colloquium Comput. Complex.* TR96 (1995)

[37] Knill, Emanuel ; Laflamme, Raymond ; Zurek, Wojciech H.: Resilient Quantum Computation. In: *Science* 279 (1998), Nr. 5349, S. 342–345. `http://dx.doi.org/10.1126/science.279.5349.342`. – DOI 10.1126/science.279.5349.342

[38] Lee, Joonho ; Huggins, William J. ; Head-Gordon, Martin ; Whaley, K. B.: Generalized Unitary Coupled Cluster Wave functions for Quantum Computation. In: *Journal of Chemical Theory and Computation* 15 (2019), Nr. 1, 311-324. `http://dx.doi.org/10.1021/acs.jctc.8b01004`. – DOI 10.1021/acs.jctc.8b01004

[39] Light, J. C. ; T. Carrington, Jr.: *Discrete-Variable Representations and Their Utilization.* Adv. Chem. Phys. 114, 263, 2000

[40] Lill, J. V. ; Parker, G. A. ; Light, J. C.: *Discrete Variable Representations and Sudden Models in Quantum Scattering.* Theory, Chem. Phys. Lett. 89, 483, 1982

[41] Martin, R.: *Electronic Structure*. Cambridge University Press, 2004

[42] *McKinsey Digital*. https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/quantum-computing-funding-remains-strong-but-talent-gap-raises-concern, 2022. – Accessed: 2023-01-27

[43] Nielsen, Michael A. ; Chuang, Isaac L.: *Quantum Computation and Quantum Information*. 10. anniversary edition. Cambridge University Press, 2000

[44] O'Malley, P. J. J. ; Babbush, R. ; Kivlichan, I. D. ; Romero, J. ; McClean, J. R. ; Barends, R. ; Kelly, J. ; Roushan, P. ; Tranter, A. ; Ding, N. ; Campbell, B. ; Chen, Y. ; Chen, Z. ; Chiaro, B. ; Dunsworth, A. ; Fowler, A. G. ; Jeffrey, E. ; Lucero, E. ; Megrant, A. ; Mutus, J. Y. ; Neeley, M. ; Neill, C. ; Quintana, C. ; Sank, D. ; Vainsencher, A. ; Wenner, J. ; White, T. C. ; Coveney, P. V. ; Love, P. J. ; Neven, H. ; Aspuru-Guzik, A. ; Martinis, J. M.: Scalable Quantum Simulation of Molecular Energies. In: *Phys. Rev. X* 6 (2016), Jul, 031007. http://dx.doi.org/10.1103/PhysRevX.6.031007. – DOI 10.1103/PhysRevX.6.031007

[45] Parr, R.G. ; Weitao, Y.: *Density-Functional Theory of Atoms and Molecules*. Oxford University Press; . ISBN 9780197560709. doi:10.1093/oso/9780195092769.001.0001., 1995

[46] Peruzzo, Alberto ; McClean, Jarrod ; Shadbolt, Peter ; Yung, Man-Hong ; Zhou, Xiao-Qi ; Love, Peter J. ; Aspuru-Guzik, Alán ; O'Brien, Jeremy L.: *A variational eigenvalue solver on a photonic quantum processor*. Nat Commun 5, 4213. https://doi.org/10.1038/ncomms5213, 2014

[47] Preskill, John: *Quantum Computing in the NISQ era and beyond*. Quantum 2, 79 https://doi.org/10.22331/q-2018-08-06-79, 2018

[48] *Qiskit: An Open-source Framework for Quantum Computing*

[49] Reiher, Markus ; Wiebe, Nathan ; Svore, Krysta M. ; Wecker, Dave ; Troyer, Matthias: *Elucidating reaction mechanisms on quantum computers*. https://doi.org/10.1073/pnas.1619152114. Version: 2017

[50] Research, Precedence: *Quantum Computing Market*. https://www.precedenceresearch.com/quantum-computing -market, 2021. – Accessed: 2023-01-27

[51] Ross, I. G.: *alculations of the energy levels of acetylene by the method of antisymmetric molecular orbitals, including σ-π interaction*. Transactions of the Faraday Society. The Royal Society of Chemistry. 48: 973–991. doi:10.1039/TF9524800973., 1952

[52] Schollwöck, U.: The density-matrix renormalization group. In: *Reviews of Modern Physics* 77 (2005), apr, Nr. 1, 259–315. http://dx.doi.org/10.1103/revmodphys.77.259. – DOI 10.1103/revmodphys.77.259

[53] Shor, Peter ; Calderbank, A. R.: *Good quantum error-correcting codes exis.* Physical Review A, Volume 54, Number 2, 1995

[54] Shor, Peter W.: *Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer.* arXiv:quant-ph/9508027, 1995

[55] Tang, Ho L. ; Shkolnikov, V.O. ; Barron, George S. ; Grimsley, Harper R. ; Mayhall, Nicholas J. ; Barnes, Edwin ; Economou, Sophia E.: Qubit-ADAPT-VQE: An Adaptive Algorithm for Constructing Hardware-Efficient Ansätze on a Quantum Processor. In: *PRX Quantum* 2 (2021), Apr, 020310. http://dx.doi.org/10.1103/PRXQuantum.2.020310. – DOI 10.1103/PRXQuantum.2.020310

[56] Thiessen, Leander: *Antisymmetrization Circuit for VQE.* https://github.com/LeanderThiessen/antisymmetrization-circuit.git, 2023

[57] Tilly, Jules ; Chen, Hongxiang ; Cao, Shuxiang ; Picozzi, Dario ; Setia, Kanav ; Li, Ying ; Grant, Edward ; Wossnig, Leonard ; Rungger, Ivan ; Booth, George H. ; Tennyson, Jonathan: *The Variational Quantum Eigensolver: A review of methods and best practices.* Physics Reports Vol 986 https://doi.org/10.1016/j.physrep.2022.08.003, 2022

[58] Wootters, W. ; Zurek, W.: *A single quantum cannot be cloned.* Nature 299, 802–803 https://doi.org/10.1038/299802a0, 1982

[59] Zhang, Feng ; Gomes, Niladri ; Berthusen, Noah F. ; Orth, Peter P. ; Wang, Cai-Zhuang ; Ho, Kai-Ming ; Yao, Yong-Xin: Shallow-circuit variational quantum eigensolver based on symmetry-inspired Hilbert space partitioning for quantum chemical calculations. In: *Phys. Rev. Res.* 3 (2021), Jan, 013039. http://dx.doi.org/10.1103/PhysRevResearch.3.013039. – DOI 10.1103/PhysRevResearch.3.013039

A Calculations

A.1 Jordan-Wigner Anti-Commutation Relations

Under the Jordan-Wigner transformation, the fermionic creation and annihilation operators $a_i^\dagger$ and a_i are mapped to the qubit operators

$$s_i^+ = \left(\prod_{k=1}^{i-1} Z_k\right) \sigma_i^- \qquad s_i^- = \left(\prod_{k=1}^{i-1} Z_k\right) \sigma_i^+ . \tag{138}$$

Claim: $\{s_i^+, s_j^+\} = \{s_i^-, s_j^-\} = 0$ and $\{s_i^+, s_j^-\} = \delta_{ij}$

Proof. Consider first the case $i \neq j$. W.l.o.g we assume that $i < j$.

$$\begin{aligned}
\{s_i^+, s_j^+\} = s_i^+ s_j^+ + s_j^+ s_i^+ &= \left(\prod_{k=1}^{i-1} Z_k\right) \sigma_i^- \cdot \left(\prod_{k=1}^{j-1} Z_k\right) \sigma_j^- + \left(\prod_{k=1}^{j-1} Z_k\right) \sigma_j^- \cdot \left(\prod_{k=1}^{i-1} Z_k\right) \sigma_i^- \\
&= \left(\sigma_i^- Z_i\right) Z_{i+1} \ldots Z_{j-1} \cdot \sigma_j^- + \left(Z_i \sigma_i^-\right) Z_{i+1} \ldots Z_{j-1} \cdot \sigma_j^- \\
&= \left(\sigma_i^- Z_i + Z_i \sigma_i^-\right) Z_{i+1} \ldots Z_{j-1} \cdot \sigma_j^- \\
&= \left(-\sigma_i^- + \sigma_i^-\right) Z_{i+1} \ldots Z_{j-1} \cdot \sigma_j^- \\
&= 0
\end{aligned}$$

The same reasoning can be used to show that $\{s_i^-, s_j^-\} = 0$ and $\{s_i^+, s_j^-\} = 0$. Let now $i = j$.

$$\begin{aligned}
\{s_i^\pm, s_i^\pm\} = 2 s_i^+ s_i^+ &= 2 \left(\prod_{k=1}^{i-1} Z_k\right) \sigma_i^\mp \cdot \left(\prod_{k=1}^{i-1} Z_k\right) \sigma_i^\mp \\
&= 2 \sigma_i^\mp \sigma_i^\mp \\
&= 0 \\
\{s_i^+, s_i^-\} &= s_i^+ s_i^- + s_i^- s_i^+ \\
&= \sigma_i^- \sigma_i^+ + \sigma_i^+ \sigma_i^- \\
&= |0\rangle \langle 1|1\rangle \langle 0|_i + |1\rangle \langle 0|0\rangle \langle 1|_i \\
&= |0\rangle \langle 0|_i + |1\rangle \langle 1|_i \\
&= \mathbb{1}
\end{aligned}$$

□

A.2 Probability of Success in the Repetition Check

In the following, we will compute the probability of successfully projecting the state of `seed` in equation (83) onto the repetition-free subspace $\mathcal{F}$. Since the application of a sorting network is unitary and does not affect the occurrence of repetitions, it does not change the probability of success. It is therefore easier to compute the probability p_{succ} of projecting the state (82) onto $\mathcal{F}$. We can rewrite the state before the projection as

$$\frac{1}{\sqrt{2^{mn'}}}\left(\sum_{|s\rangle\in\mathcal{F}}|s\rangle+\sum_{|t\rangle\in\mathcal{F}^{\perp}}|t\rangle\right). \tag{139}$$

The probability of success can thus be determined by counting the number of states in $\mathcal{F}$ and $\mathcal{F}^{\perp}$. The number of length-m strings of numbers $s_i \in [0, 2^{n'}-1]$ without repetitions is given by

$$m(m-1)\cdot\ldots\cdot(2^{n'}-m+1)=\frac{m!}{(2^{n'}-m)!}. \tag{140}$$

The probability of success is equal to this number divided by the total number of length-m strings:

$$p_{\text{succ}}=\frac{1}{2^{mn'}}\frac{m!}{(2^{n'}-m)!}. \tag{141}$$

Proposition A.01 in [11] can be used to bound p_{succ} in the following way:

$$p_{\text{succ}}\geq 1-\frac{m(m-1)}{2^{n'+1}}. \tag{142}$$

A.3 Disentangling `seed` and `record`

As described in [8], it is necessary to show that sorting `seed` does not generate entanglement with `record`. The joint state of `seed` and `record` after projecting onto $\mathcal{F}$ may be written as

$$\frac{1}{\sqrt{2^{mn'}}}\sum_{s_1<\ldots<s_m}\sum_{\sigma\in S_m}|\sigma(s_1\ldots s_m)\rangle_{\texttt{seed}}\,|\imath\rangle_{\texttt{record}}\,, \tag{143}$$

where $\imath$ represents the identity permutation. The application of a sorting network in step 2 corresponds to performing a sequence of permutations $\sigma_1,\ldots,\sigma_L$, such that

$$\sigma_L \circ ... \circ \sigma_1 \circ \sigma(s_1, ..., s_m) = (s_1, ..., s_m) \quad \forall\, \sigma \in S_m\,. \tag{144}$$

When applied to equation (143), each permutation σ_i changes one qubit $|\sigma_i\rangle$ in `record`. After L permutations, the joint state of `seed` and `record` is given by

$$\frac{1}{\sqrt{2^{mn'}}} \sum_{s_1<...<s_m} |s_1...s_m\rangle_{\texttt{seed}} \sum_{\sigma\in S_m} |\sigma_1...\sigma_L\rangle_{\texttt{record}}\,. \tag{145}$$

It is important to note that we can only write this state as a product state because the `seed` does not contain repetitions. If it did, the representation (143) would not be possible.

A.4 Matrix Elements in the Plane Wave Basis

In the following, we derive the expressions for the matrix elements of T, U, and V in equations (90) and (91).

i) Kinetic Energy T_{pq}

$$T_{\boldsymbol{pq}} = \int_\Omega \mathrm{d}\boldsymbol{r}\, \phi^*_{\boldsymbol{p}}(\boldsymbol{r}) \left(-\frac{\nabla^2}{2}\right) \phi_{\boldsymbol{q}}(\boldsymbol{r}) \tag{146}$$

$$= -\frac{1}{2\Omega} \int_\Omega \mathrm{d}\boldsymbol{r}\, e^{i\boldsymbol{k_p r}} \nabla^2 e^{-i\boldsymbol{k_q r}} \tag{147}$$

$$= \frac{\|\boldsymbol{k_q}\|^2}{2\Omega} \int_\Omega \mathrm{d}\boldsymbol{r}\, e^{i(\boldsymbol{k_p}-\boldsymbol{k_q})\boldsymbol{r}} \tag{148}$$

$$= \frac{\|\boldsymbol{k_q}\|^2}{2\Omega} \int_{-\Omega^{1/3}/2}^{\Omega^{1/3}/2} \mathrm{d}x e^{ik^x_{\boldsymbol{p}-\boldsymbol{q}}x} \int_{-\Omega^{1/3}/2}^{\Omega^{1/3}/2} \mathrm{d}y e^{ik^y_{\boldsymbol{p}-\boldsymbol{q}}y} \int_{-\Omega^{1/3}/2}^{\Omega^{1/3}/2} \mathrm{d}z e^{ik^z_{\boldsymbol{p}-\boldsymbol{q}}z} \tag{149}$$

$$= \begin{cases} \frac{\|\boldsymbol{k_q}\|^2}{2} & \text{if} \quad \boldsymbol{k_{p-q}} = 0 \\ \frac{\|\boldsymbol{k_q}\|^2}{\Omega} \frac{1}{k^x_{\boldsymbol{p}-\boldsymbol{q}}} \frac{1}{k^y_{\boldsymbol{p}-\boldsymbol{q}}} \frac{1}{k^z_{\boldsymbol{p}-\boldsymbol{q}}} \underbrace{\sin\big(\frac{k^x_{\boldsymbol{p}-\boldsymbol{q}}\Omega^{1/3}}{2}\big)}_{\sin(\pi(p_x-q_x))=0} \sin\big(\frac{k^y_{\boldsymbol{p}-\boldsymbol{q}}\Omega^{1/3}}{2}\big) \sin\big(\frac{k^z_{\boldsymbol{p}-\boldsymbol{q}}\Omega^{1/3}}{2}\big) = 0 & \text{if} \quad \boldsymbol{k_{p-q}} \neq 0 \end{cases}$$

In line (148) we have used the fact that $\boldsymbol{k_p} - \boldsymbol{k_q} = \boldsymbol{k_{p-q}}$. The key observation is that

each integral in line (149) is only non-zero if $k^i_{p-q} = 0$. By evaluating the expression for $\boldsymbol{k_{p-q}} = 0$ we arrive at an expression for the matrix element.

$$T_{\boldsymbol{pq}} = \frac{\|\boldsymbol{k_q}\|^2}{2}\delta_{\boldsymbol{pq}} \tag{150}$$

ii) External Potential Energy U_{pq}

Next, we compute the matrix elements $U_{\boldsymbol{pq}}$ of the external energy operator U describing the interaction between electrons and fixed nuclei.

$$U_{\boldsymbol{pq}} = \int_\Omega \mathrm{d}\boldsymbol{r}\, \phi^*_{\boldsymbol{p}}(\boldsymbol{r}) \left(-\sum_{l=1}^{M} \frac{Z_l}{|\boldsymbol{r} - \boldsymbol{R}_l|} \right) \phi_{\boldsymbol{q}}(\boldsymbol{r}) \tag{151}$$

$$= \frac{1}{\Omega} \sum_{l=1}^{M} Z_l \int_\Omega \mathrm{d}\boldsymbol{r}\, \frac{e^{i\boldsymbol{k_{p-q}}\boldsymbol{r}}}{|\boldsymbol{r} - \boldsymbol{R}_l|} \tag{152}$$

$$= \frac{1}{\Omega} \sum_{l=1}^{M} Z_l e^{i\boldsymbol{k_{p-q}}\boldsymbol{R}_l} \int_{\Omega'} \mathrm{d}\boldsymbol{r}'\, \frac{e^{i\boldsymbol{k_{p-q}}\boldsymbol{r}'}}{|\boldsymbol{r}'|} \tag{153}$$

Here, we have used a substitution $\boldsymbol{r}' = \boldsymbol{r} - \boldsymbol{R}_l$. In order to evaluate the integral on the right hand side, we observe that it corresponds exactly to the Fourier transform of the Coulomb potential in the periodic picture. In particular, we define the non-periodic potential $V(\boldsymbol{r})$ and the periodic potential $V_{\mathrm{per}}(\boldsymbol{r})$, such that

$$V(\boldsymbol{r}) := \frac{1}{|\boldsymbol{r}|} \qquad \text{and} \qquad V_{\mathrm{per}}(\boldsymbol{r}) := \sum_{\boldsymbol{p}\in\mathbb{Z}} \frac{1}{|\boldsymbol{r} - \boldsymbol{p}\Omega^{1/3}|}\,. \tag{154}$$

$V_{\mathrm{per}}(\boldsymbol{r})$ describes the scenario we consider in section 4.3, i.e. the configuration where a copy of the molecule is placed in the center of each lattice cell. The Fourier transform of the non-periodic potential $V(\boldsymbol{r})$ can then be evaluated by simply integrating the periodic potential $V_{\mathrm{per}}(\boldsymbol{r})$ over one lattice cell.

$$\mathscr{F}\left[V\right](\boldsymbol{k}) = \int_{\mathbb{R}^3} \mathrm{d}\boldsymbol{r}\, e^{i\boldsymbol{k}\boldsymbol{r}} V(\boldsymbol{r}) = \int_\Omega \mathrm{d}\boldsymbol{r}\, e^{i\boldsymbol{k}\boldsymbol{r}} V_{\mathrm{per}}(\boldsymbol{r}) \tag{155}$$

Intuitively, integrating the periodic potential over a single lattice cell includes contributions of all other copies of the potential located in different lattice cells. In the case of the Coulomb potential, the Fourier transform is equal to $\mathscr{F}\left[V\right](\boldsymbol{k}) = 4\pi/\|\boldsymbol{k}\|^2$. By inserting

this into the expression for $U_{\boldsymbol{pq}}$ we therefore find

$$U_{\boldsymbol{pq}} = \frac{1}{\Omega} \sum_{l=1}^{M} Z_l \frac{e^{i\boldsymbol{k_{p-q}}\boldsymbol{R}_l}}{\|\boldsymbol{k_{p-q}}\|^2} \tag{156}$$

ii) Interaction Energy V_{pqrs}

$$V_{\boldsymbol{pqrs}} = \int_\Omega \mathrm{d}\boldsymbol{r}_1 \mathrm{d}\boldsymbol{r}_2 \frac{\phi^*_{\boldsymbol{p}}(\boldsymbol{r}_1)\, \phi^*_{\boldsymbol{r}}(\boldsymbol{r}_2)\, \phi_{\boldsymbol{q}}(\boldsymbol{r}_2)\, \phi_{\boldsymbol{s}}(\boldsymbol{r}_1)}{|\boldsymbol{r}_1 - \boldsymbol{r}_2|} \tag{157}$$

$$= \frac{1}{\Omega^2} \int_\Omega \mathrm{d}\boldsymbol{r}_1 \mathrm{d}\boldsymbol{r}_2 e^{i\boldsymbol{k_{p-s}}\boldsymbol{r}_1} e^{i\boldsymbol{k_{r-q}}\boldsymbol{r}_2} \frac{1}{|\boldsymbol{r}_1 - \boldsymbol{r}_2|} \tag{158}$$

$$= \frac{1}{\Omega^2} \int_\Omega \mathrm{d}\boldsymbol{r}' \mathrm{d}\boldsymbol{r}_2 e^{i\boldsymbol{k_{p-s}}\boldsymbol{r}'} e^{i\boldsymbol{k_{p-s}}\boldsymbol{r}_2} e^{i\boldsymbol{k_{r-q}}\boldsymbol{r}_2} \frac{1}{|\boldsymbol{r}'|} \tag{159}$$

$$= \frac{1}{\Omega^2} \underbrace{\int_\Omega \mathrm{d}\boldsymbol{r}' \frac{e^{i\boldsymbol{k_{p-s}}\boldsymbol{r}'}}{|\boldsymbol{r}'|}}_{\mathscr{F}[V](\boldsymbol{k_{p-s}})} \int_\Omega \mathrm{d}\boldsymbol{r}_2 e^{i(\boldsymbol{k_{p-s}}+\boldsymbol{k_{r-q}})\boldsymbol{r}_2} \tag{160}$$

Again, using the Fourier transform of the Coulomb non-periodic Coulomb potential, we arrive at an expression for the matrix element

$$V_{\boldsymbol{pqrs}} = \frac{4\pi}{\|\boldsymbol{k_{p-s}}\|^2} \delta_{\boldsymbol{p-s},\boldsymbol{q-r}}\,. \tag{161}$$

A.5 Analytic Form of the Plane Wave Dual Basis

In this section, we derive an analytical expression of the wavefunction component $\tilde{\phi}_{p_x}(x)$ of a basis vector in the plane wave dual basis. Starting from equation (96), we write

$$\widetilde{\phi}_{p_x}(x) = \sqrt{\frac{1}{N^{1/3}}} \sum_{\nu_x=-N^{1/3}/2}^{N^{1/3}/2} e^{-2\pi i p_x \nu_x / N^{1/3}} \phi_{\nu_x}(x) \tag{162}$$

$$= \sqrt{\frac{1}{(\Omega N)^{1/3}}} \sum_{\nu_x=-N^{1/3}/2}^{N^{1/3}/2} e^{-2\pi i p_x \nu_x / N^{1/3}} e^{2\pi i \nu_x x / \Omega^{1/3}} \tag{163}$$

$$= \sqrt{\frac{1}{(\Omega N)^{1/3}}} \sum_{\nu_x=-N^{1/3}/2}^{N^{1/3}/2} e^{2\pi i \nu_x \overbrace{\left(\frac{x}{\Omega^{1/3}} - \frac{p_x}{N^{1/3}}\right)}^{\bar{x}}} \tag{164}$$

$$= \sqrt{\frac{1}{(\Omega N)^{1/3}}} \left(\sum_{\nu_x=0}^{N^{1/3}/2} e^{2\pi i \nu_x \bar{x}} + \sum_{\nu_x=0}^{N^{1/3}/2} e^{-2\pi i \nu_x \bar{x}} + 1 \right) \tag{165}$$

$$= \sqrt{\frac{1}{(\Omega N)^{1/3}}} \left(\frac{1 - e^{i\pi\bar{x}(N^{1/3}+\frac{1}{2})}}{1 - e^{2\pi i \bar{x}}} + \frac{1 - e^{-i\pi\bar{x}(N^{1/3}+\frac{1}{2})}}{1 - e^{-2\pi i \bar{x}}} - 1 \right) \tag{166}$$

$$= \sqrt{\frac{1}{(\Omega N)^{1/3}}} \left(\frac{1}{\sin(\pi\bar{x})} \Big(\sin(\pi\bar{x}) + \sin\Big(\pi\bar{x}(N^{1/3} - 1/2)\Big) \Big) - 1 \right) \tag{167}$$

$$= \sqrt{\frac{1}{(\Omega N)^{1/3}}} \frac{\sin\big(\pi\bar{x}(N^{1/3} - 1/2)\big)}{\sin(\pi\bar{x})} \tag{168}$$

$$= \sqrt{\frac{1}{(\Omega N)^{1/3}}} \frac{\sin\left(\frac{\pi(N^{1/3}-1/2)}{\Omega^{1/3}} x - \pi p_x (1 - \frac{N^{-1/3}}{2})\right)}{\sin\left(\frac{\pi}{\Omega^{1/3}} x - \frac{\pi p_x}{N^{1/3}}\right)} \tag{169}$$

In equation (166) we have used the geometric series $\sum_{k=0}^{N-1} q^k = (1-q)^N/(1-q)$. The full wavefunction $\widetilde{\phi}_{\boldsymbol{p}}(\boldsymbol{r})$ in equation (98) can be obtained by multiplying the spatial components.

A.6 Bounds for the Measurement Scaling in the Plane Wave Basis

To derive the bound in equation (123), we use the representation of T in the plane wave basis (92).

$$\max_{\psi} |\langle\psi| T |\psi\rangle| = \max_{\psi} \left| \langle\psi| \left(\frac{1}{2} \sum_{i=1}^{m} \sum_{\boldsymbol{p}} \|\boldsymbol{k}_{\boldsymbol{p}}\|^2 |\boldsymbol{p}\rangle \langle\boldsymbol{p}|_i \right) |\psi\rangle \right| \tag{170}$$

$$\leq \max_{\psi} \frac{1}{2} \sum_{i=1}^{m} \sum_{\boldsymbol{p}} \left\| \frac{2\pi \boldsymbol{p}}{\Omega^{1/3}} \right\|^2 \langle\psi| \left(|\boldsymbol{p}\rangle \langle\boldsymbol{p}|_i \right) |\psi\rangle \tag{171}$$

$$\leq \max_{\psi} \sum_{i=1}^{m} \frac{1}{2} \left(\frac{2\pi N^{1/3}}{\Omega^{1/3}} \right)^2 \langle\psi| \underbrace{\sum_{\boldsymbol{p}} |\boldsymbol{p}\rangle \langle\boldsymbol{p}|_i}_{\mathbb{1}_i} |\psi\rangle \tag{172}$$

$$\leq \frac{\pi m N^{2/3}}{2\Omega^{2/3}} \tag{173}$$

The constant factor in (123) is thus given by $C_T = \pi/2$. Similarly we express the maximum expectation value of V as

$$\max_{\psi} |\langle\psi| V |\psi\rangle| = \max_{\psi} \left| \langle\psi| \frac{2\pi}{\Omega} \sum_{i \neq j} \sum_{\boldsymbol{\nu} \neq 0} \sum_{\tilde{\boldsymbol{p}}, \tilde{\boldsymbol{q}}} \frac{\cos\left(\boldsymbol{k}_{\boldsymbol{\nu}} \boldsymbol{r}_{\tilde{\boldsymbol{p}} - \tilde{\boldsymbol{q}}}\right)}{\|\boldsymbol{k}_{\boldsymbol{\nu}}\|^2} |\tilde{\boldsymbol{p}}\rangle \langle\tilde{\boldsymbol{p}}|_i |\tilde{\boldsymbol{q}}\rangle \langle\tilde{\boldsymbol{q}}|_j |\psi\rangle \right| \tag{174}$$

$$\leq \max_{\psi} \frac{2\pi}{\Omega} \sum_{i \neq j} \sum_{\boldsymbol{\nu} \neq 0} \sum_{\tilde{\boldsymbol{p}}, \tilde{\boldsymbol{q}}} \frac{\left|\cos\left(\boldsymbol{k}_{\boldsymbol{\nu}} \boldsymbol{r}_{\tilde{\boldsymbol{p}} - \tilde{\boldsymbol{q}}}\right)\right|}{\|\boldsymbol{k}_{\boldsymbol{\nu}}\|^2} \langle\psi| \left(|\tilde{\boldsymbol{p}}\rangle \langle\tilde{\boldsymbol{p}}|_i |\tilde{\boldsymbol{q}}\rangle \langle\tilde{\boldsymbol{q}}|_j \right) |\psi\rangle \tag{175}$$

$$\leq \max_{\psi} \frac{2\pi}{\Omega} \sum_{i \neq j} \left(\sum_{\boldsymbol{\nu} \neq 0} \frac{1}{\|\boldsymbol{k}_{\boldsymbol{\nu}}\|^2} \right) \langle\psi| \underbrace{\sum_{\tilde{\boldsymbol{p}}, \tilde{\boldsymbol{q}}} |\tilde{\boldsymbol{p}}\rangle \langle\tilde{\boldsymbol{p}}|_i |\tilde{\boldsymbol{q}}\rangle \langle\tilde{\boldsymbol{q}}|_j}_{\mathbb{1}_i \otimes \mathbb{1}_j} |\psi\rangle \tag{176}$$

$$\leq \frac{2\pi m^2}{\Omega} \underbrace{\left(\sum_{\boldsymbol{\nu} \neq 0} \frac{1}{\|\boldsymbol{k}_{\boldsymbol{\nu}}\|^2} \right)}_{i)} \tag{177}$$

The sum $i)$ cannot be solved analytically, but it can be shown [8] it scales with the the number of orbitals as $\mathcal{O}(N^{1/3})$. The maximum expectation value of V therefore exhibits the scaling $\mathcal{O}(m^2 N^{1/3} \Omega^{-1/3})$.

A.7 UCCSD Ansatz in First Quantization

The Unitary Coupled Cluster Ansatz (UCC) with single and double excitations is given by the unitary

$$U = e^{\sum_{ia} \hat{\tau}_i^a + \sum_{ijab} \hat{\tau}_{ij}^{ab}} . \tag{178}$$

In order to implement this operator in terms of elementary quantum gates, it is necessary to represent it as a product of Pauli string exponentials. Since the terms in the exponential (178) generally do not commute, we need to use a Trotter expansion. As described in section 3.3.1, it can be shown numerically that even choosing just a single Trotter step is usually sufficient.

To estimate the required gate depth, we first need to determine the number of Pauli strings in the expression in equation (178). Since we are only interested in the scaling behaviour, we focus on the double-excitation operators $\hat{\tau}_{ij}^{ab}$ and ignore the single-excitation operators $\hat{\tau}_i^a$. In Second Quantization, $\hat{\tau}_{ij}^{ab}$ can be expanded as a sum of $8 = \mathcal{O}(1)$ Pauli strings. Summing over all indices i, j, a, and b thus results in $\mathcal{O}(N^4)$ Pauli strings. Under a Trotter approximation we write this as a product of $\mathcal{O}(N^4)$ Pauli string exponentials. As shown in section 4.4.1, each of these Pauli string exponentials requires a gate depth of $\mathcal{O}(N)$ when using the Jordan-Wigner or Parity encoding, and a gate depth of $\mathcal{O}(\log N)$ when using the Bravyi-Kitaev encoding. In the best case, the unitary (178) can thus be implemented with a gate depth of $\mathcal{O}(N^4 \log N)$.

In First Quantization, each term $\hat{\tau}_{ij}^{ab}$ is expanded as a sum of $\mathcal{O}(N^2)$ Pauli strings. Summing over the indices i, j, a, and b thus results in $\mathcal{O}(N^6)$ Pauli strings. Again, each Pauli string exponential can be implemented using a gate depth of $\mathcal{O}(\log N)$, leading to a total gate depth for the operator U of $\mathcal{O}(N^6 \log N)$.

The UCCSD Ansatz in this form is therefore not suitable for VQE in First Quantization.

A.8 Antisymmetry Conservation

In the following, we provide a simple example showing that operators of the form $e^{i\theta P}$ in the qubit-ADAPT-VQE Ansatz do not conserve antisymmetry when acting on an antisymmetric reference state.

Consider a system of $m = 2$ electrons and $N = 4$ orbitals in the reference state

$$|\psi\rangle = \frac{1}{\sqrt{2}} \left(|\phi_0\rangle |\psi_3\rangle - |\phi_3\rangle |\psi_0\rangle \right) \tag{179}$$

$$= \frac{1}{\sqrt{2}} \left(|00\rangle |11\rangle - |11\rangle |00\rangle \right) \tag{180}$$

Let $P = X_1X_2$ be a Pauli string acting on the first electron. The Ansatz operator $e^{i\theta P}$ can then be expressed as the operator $e^{i\theta X_1X_2} = \cos(\theta)\mathbb{1}_{12} + i\sin(\theta)X_1X_2$.

$$e^{i\theta P}|\psi\rangle = (\cos(\theta)\mathbb{1}_{12} + i\sin(\theta)X_1X_2)\frac{1}{\sqrt{2}}(|00\rangle|11\rangle - |11\rangle|00\rangle) \tag{181}$$

$$= \cos(\theta)|00\rangle|11\rangle + i\sin(\theta)|11\rangle|11\rangle - \cos(\theta)|11\rangle|00\rangle - i\sin(\theta)|00\rangle|00\rangle \tag{182}$$

$$= \cos(\theta)|\phi_0\rangle|\phi_3\rangle + i\sin(\theta)|\phi_3\rangle|\phi_3\rangle - \cos(\theta)|\phi_3\rangle|\phi_0\rangle - i\sin(\theta)|\phi_0\rangle|\phi_0\rangle \tag{183}$$

This state is only antisymmetric if $\sin(\theta) = 0$, which is evidently not the case for arbitrary θ. In particular, (183) is a superposition that contains both ordered and non-ordered states, as well as repetition-free states and states with repetitions.

B Circuits

B.1 Antisymmetrization Circuit for $m = 2$

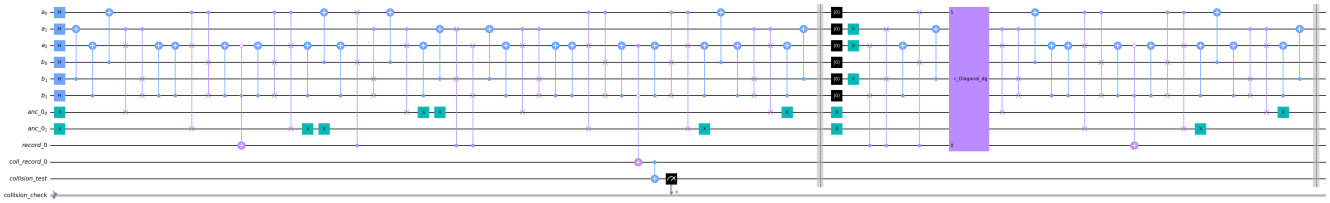


FIGURE 27: The circuit from Fig. 20 for $m = 2$ electrons and $N = 8$ basis functions without the inclusion of barriers. Note that some gates cancel.

B.2 Antisymmetrized Ansatz Circuit $m = 2$