Hamiltonian Simulation with the Spectrum by Quantum Walk Algorithm

Author
Floris van den Ende
Amsterdam University College
florisvdende@gmail.com

Supervisor
Dhr. Dr. J.van Wezel
Faculteit der Natuurwetenschappen en
Informatica & QuSoft
j.vanwezel@uva.nl

Reader
Christian Majenz
QuSoft
christian.majenz@cwi.nl

Supervisor
Joris Kattemölle
QuSoft
j.j.kattemolle@uva.nl

Major: Sciences

June 10, 2020

6000 words

A
Amsterdam university college
UC

Abstract

Transitionary sentence of qchem. A fundamental problem in chemistry is finding the ground state and ground energy of a molecule, called the electronic structure problem. While classical computers struggle with solving large, highly correlated systems due to the exponentially growing complexity, quantum computers are able to efficiently simulate quantum systems, which can lead to an increased understanding of complex chemical reactions. If quantum computers succeed in providing insight in nitrogenase, the world's total energy output may be significantly reduced. But despite rapid advancements in all facets of quantum computing, new theoretical advancements are needed to solve the electronic structure problem for complex systems in the future.

We discussed the Spectrum by quantum walk algorithm proposed by Poulin et al. [1], where the main idea is to "simulate not imitate" [1, p. 5]. While methods such as Trotterization attempt to approximate the time evolution operator $U(t)$, this algorithm instead employs another, exactly implementable unitary operator W which is a function of the Hamiltonian. In Chapter 3, we have provided proofs of the mathematical properties of the operations that W consists of and we have shown the relation of the spectrum of the Hamiltonian with the spectrum of W. Energy measurements can be performed

Contents

Chapter 1

Introduction

Quantum computing is one of the most promising emerging scientific and technological fields of the past decades [2]. The theoretical computational advantages of quantum computing allow for valuable applications in fields such as cryptography, chemistry and machine learning. However, building useful quantum computers is highly complicated: the fundamental issue is to perform high-fidelity operations on a coherent, scalable set of qubits [3]. Peter Shor's prime factorization algorithm first showed the potential of quantum algorithms and in the years since, the field of quantum computing has rapidly expanded and can be divided into many, interdisciplinary sub-fields [4]. Among these fields is quantum computational chemistry, which aims to solve classically intractable chemistry problems. Progress within this field might elucidate unresolved problems of solid-state physics, superconductivity at high temperature and some complex chemical reactions [2]. However, as the physical implementation of quantum computers is still severely limited by issues such as decoherence and quantum noise, quantum algorithms need to be optimized to lower the number of gates needed to perform the desired calculations. While quantum algorithms have a range of capabilities within quantum computational chemistry, we focus on one problem in particular, called the *electronic structure problem*: finding the ground state and ground energy of chemical systems. We study this specific problem because knowledge of the energy eigenstates allows predictions of other fundamental properties of the system, such as reaction rates and location of stable structures [2, p. 3]. Furthermore, we can relatively easily transfer the algorithms and methods used for the electronic structure problem to other problems such as determining the transition states.

There are several methods to classically solve the *electronic structure problem*, such as density functional theory and the Hartree-Fock method. While these calculations are highly efficient, they struggle with large, highly correlated systems, due to the exponential complexity. McArdle et al. [2] state that there is a knowledge gap of systems of about 100-200 spin-orbitals, as such systems are too large and too correlated to accurately calculate with classical methods. Within this range falls the molecule FeMoCo, having 113 electrons in 152 spin-orbitals.
To give some context as to why this molecule is of special interest, it is important to note that three percent of the world's energy output is spent on making fertilizer [5]. We currently rely on a highly energy-intensive process requiring very large amounts of natural gas. However, there exist anaerobic bacteria performing a process serving the exact same function while requiring much less energy, utilizing nitrogen fixation with molecules such as FeMoCo [2]. Traditional chemical analyses have not been able to provide an understanding on the details of this process, and, as this molecule is highly complex, simulations using classical supercomputers are out of reach. Reiher et al. [5] have shown that quantum computers are, even when taking in account the substantial decoherence and error of current gate computations, able to provide new insights on chemical processes such as nitrogen fixation in nitrogenase. However, a resource estimation performed by Berry et al. [6] shows that the resources needed to simulate FeMoCo is equivalent to roughly one million qubits running for 2 months. While this result has been an improvement over previous estima-

tions, it is clear that it is still out of reach currently [2]. But if the scientific community succeeds in obtaining a greater understanding of nitrogenase, new and more efficient processes of creating fertilizer may be developed, significantly reducing the global carbon footprint. This thesis aims to contribute towards minimizing gate count by exploring an algorithm which may provide speedup in solving the electronic structure problem.

The general set of techniques used to simulate a quantum system with a quantum computer is called *Hamiltonian simulation* [7]. Perhaps the most used Hamiltonian simulation method is the Variational Quantum Eigensolver (VQE). The VQE prepares a trial wave function and approximates the energy by iteratively retrieving the Pauli operator tensor products. This algorithm is often used on hybrid computers, as the optimization algorithms required are much faster on a classical computer than a quantum computer. While VQE is a very practical application for near-term quantum computers, methods using phase estimation, which will be explained in section 3.4, offer more efficient circuit implementations in terms of gate complexity [8]. The standard approach is to approximate the time-evolution operator of the Hamiltonian. However, this is prone to approximation errors. Poulin et al. [1] have recently proposed another algorithm that circumvents the need for approximating the time-evolution operator by *qubitization*, using ideas and techniques from quantum walks.

In this thesis, we will thoroughly analyze the algorithm proposed by Poulin et al. [1] and adapt an implementation of the algorithm for two molecular systems. The research process of this capstone is composed of three main components. The first component (Chapter 2) consists of a literature review of the fundamentals of quantum computing and quantum simulation, introducing standard conventions and treating concepts appearing frequently in later sections. To do so, several established texts such as Nielsen and Chuang [9], McArdle et al. [2] and Cao et al. [7] will be synthesised. The second component (Chapter 3) expands on the algorithm, as well as giving proofs of the mathematical claims made by Poulin et al. [1]. The third component (Chapter 4) proposes a circuit implementation of the algorithm and adapts the ProjectQ [1] implementation made by Steiger [10] for two Hamiltonians of H_2 and H. Finally, the results of the implementation and the prospects of this algorithm are discussed.

---

[1] http://projectq.ch

Chapter 2

Quantum Computing

In this chapter, we will provide the necessary preliminary information needed to interpret the algorithm discussed in later chapters. In the first section, an introduction to quantum computing is provided. In the second section, the general framework of quantum simulation is laid out, and more specifically, applied to the electronic structure problem.

2.1 Quantummechanical foundations

Before we delve into the specifics of quantum computing, a basic understanding of the quantum-mechanical principles governing qubits is introduced. The state of some general quantum-mechanical system is described by its wavefunction. The natural language of quantum mechanics is linear algebra and it is common practice to describe states by its wavefunction in bra-ket notation. States are described as vectors, and an arbitrary state of a 2-state quantum system can be written as

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle\,,$$

where $|\psi\rangle$ has to satisfy the normalization condition, implying that $|\alpha|^2 + |\beta|^2 = 1$. Such a system is called the *qubit*, which lives in a two-dimensional Hilbert space. The defining computational strength of qubits is that the Hilbert space grows exponentially with the number of qubits, as they can be in superposition. When measuring a qubit, the outcome of the measurement in the so-called computational basis will still be either $|0\rangle$ or $|1\rangle$, but these outcomes occur with probabilities depending on the superposition of the state, namely $|\alpha|^2$ or $|\beta|^2$. Applying quantum logic gates manipulates both basis states at the same time, resulting in an increased computational potential when compared to classical bits. The orthonormal basis states $|0\rangle$ and $|1\rangle$ are generally just an abstract representation of a two-level system, but depending on the type of quantum computer, they can represent physical properties such as charge in quantum dots as well as energy levels in trapped ions [2].

For a system with n qubits, the joint state of that system is described by taking the tensor product of each individual qubit state within that system. Taking a tensor product of two states is mathematically equivalent to taking the Kronecker product of the vectors of that state. For some arbitrary states $|\phi\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$ and $|\psi\rangle = \begin{pmatrix} \gamma \\ \delta \end{pmatrix}$ where α, β, γ and δ are complex numbers, the joint state is given by

$$|\phi\rangle \otimes |\psi\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \otimes \begin{pmatrix} \gamma \\ \delta \end{pmatrix} = \begin{pmatrix} \alpha\gamma \\ \alpha\delta \\ \beta\gamma \\ \beta\delta \end{pmatrix}$$

Analogous to classical computers, quantum computers are built from quantum circuits consisting of operations manipulating the quantum information encoded in the qubits. These operations, called *quantum gates*, have to be reversible. This places the constraint on an operation U:

$U^\dagger U = I$, implying that all quantum gates are unitary. Some of the fundamental gates are the Pauli matrices

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \tag{2.1}$$

the single qubit rotation gates

$$R_x(\theta) = \exp\left\{\frac{-i\theta X}{2}\right\}, \quad R_y(\theta) = \exp\left\{\frac{-i\theta Y}{2}\right\}, \quad R_z(\theta) = \exp\left\{\frac{-i\theta Z}{2}\right\} \tag{2.2}$$

and the Hadamard and T gates [9]

$$\mathrm{H} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \quad \mathrm{T} = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}. \tag{2.3}$$

While the above gates are all single-qubit operations, multi-qubit gates are paramount as well, as the controlled-NOT operations are used to create entangled states. Entangled states are states that can not be written as a tensor product between two other states, such as a Bell state $|\beta\rangle = \frac{|00\rangle+|11\rangle}{\sqrt{2}} = \mathrm{CNOT}_{1,2}(H \otimes I)|00\rangle$. Quantum gates are in essence rotations around the Hilbert space (which is the Bloch sphere for single qubits) and any unitary matrix may be decomposed as a series of rotations, which can be approximated to an arbitrary accuracy with just single-qubit gates and CNOT gates [2].

2.1.1   Identities

Now we will briefly discuss some nomenclature and important algebraic identities that will be used later on. Within quantum computing, indexes such as j often represent a binary bit string. For example, the bit-string of 5 is represented in vector form by the following:

$$|5\rangle = |0101\rangle = |0\rangle \otimes |1\rangle \otimes |0\rangle \otimes |1\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}.$$

Or generally, for some n in $\mathbb{N}$, the corresponding bit string state $|n\rangle$ in can be written as (while counting from zero) $\{e_n\}$, where e_n is a vector with a 1 inserted at the nth position, and zeroes elsewhere.

All bit string states are orthogonal, implying

$$\langle i|j\rangle = \delta_{i,j} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}. \tag{2.4}$$

When summing bit string projectors over the n-dimensional space, each basis vector in $\mathbb{N}^n$ has a contribution of 1, resulting in an n by n matrix with only entries of 1 on the diagonal, which is the identity matrix:

$$\sum_j^n |j\rangle\langle j| = I_n. \tag{2.5}$$

The expectation value of some observable A measured in some basis $\{\varphi, \phi\}$ is:

$$\langle \varphi| A |\phi\rangle = \langle \varphi|A\phi\rangle = \langle A^\dagger\varphi|\phi\rangle \tag{2.6}$$

However, if c is some scalar, then:

$$\langle \varphi | \, c \, | \phi \rangle = \langle \varphi | c \phi \rangle = c \, \langle \varphi | \phi \rangle \,. \tag{2.7}$$

An important tensor product identity is the following:

$$(U \otimes V)(|\psi\rangle \otimes |\phi\rangle) = (U|\psi\rangle) \otimes (V|\phi\rangle). \tag{2.8}$$

2.2   Quantum Simulation

The core objective of simulation is solving the set of differential equations that describe the physical evolution in time of that system. In the case of quantum simulation, the dynamical behaviour of the system is described by the Schrödinger equation,

$$i\hbar \frac{d}{dt} |\psi\rangle = H |\psi\rangle \,.$$

The initial observation that motivated the entire field of quantum simulation is that the storage needed to describe the wavefunction of a system of N particles grows with $\mathcal{O}(\exp\{N\})$ on a classical computer, while this relation is $\mathcal{O}(N)$ for qubits [7]. It has been shown that evolving the wavefunction in time is efficiently possible for Hamiltonians that are, to a large extent, physically realizable [7]. That is to say that the number of quantum gates needed to simulate a system grows polynomially, rather than exponentially with parameters such as system size and time.

Furthermore, it is important to note that essentially all algorithms within quantum chemistry assume error-free quantum computations. Currently, fully coherent quantum computation has not yet been realized, as the physical realizations of quantum computers are still too error prone. The process of minimizing decoherence in quantum computers is instrumental to quantum computing and is an entire own subfield of quantum computing (Quantum Error Correction), which is out of scope for this thesis. More information on this can be found in the book Quantum Computation and Quantum Information by Nielsen and Chuang [9]. The fact that quantum chemistry algorithms do not account for errors mean that none of these algorithms are executable in a nontrivial way yet, i.e. deliver results that classical computers could not have obtained. This does not mean that these algorithms cannot be implemented as we will show a rudimentary implementation of a specific quantum simulation algorithm later.

Quantum simulation can be divided into two different sets of problems, the dynamic problem and the static problem. The static problem consists of preparing the (correct) initial state and the dynamical problem consists of reproducing how the time-evolution unitary operator, $U = \exp\{-iHt\}$ affects the state of the system. Different models of systems place different constraints on the manifestation of the form of the Hamiltonian. For most physical systems of N particles, the Hamiltonian can be written as $H = \sum_k^L H_k$, where each H_k acts on a specific subsystem. Often these terms are two-body interaction terms such as $X_i X_j$ or even one-body Hamiltonians X_i. More specifically, the spin Ising model $H_{\text{Ising}} = g \sum_j X_j + J \sum_{\langle i,j \rangle} Z_i Z_j$ consists entirely of a linear combination of such interaction terms. The conventional method for simulating Hamiltonians of this form is to approximate the time-evolution operator $U(t)$ such that

$$U(t) \approx \left(e^{-iH_1 t/n} \dots e^{-iH_\ell t/n} \right)^n \,.$$

The method for approximating U as such is called *Trotter decomposition*, and, since the gatecount grows polynomially with the number of qubits and with time, this method is efficient for local-interaction Hamiltonians [7].
However, not all Hamiltonians satisfy the local-interaction condition of the Trotter decomposition, such as the electronic structure problem. The Hamiltonian of a molecule of K nuclei and N electrons is the following [2]:

$$H = -\sum_i \frac{\hbar^2}{2m_e}\nabla_i^2 - \sum_I \frac{\hbar^2}{2M_I}\nabla_I^2 - \sum_{i,I} \frac{e^2}{4\pi\epsilon_0}\frac{Z_I}{|\mathbf{r}_i - \mathbf{R}_I|}$$
$$+ \frac{1}{2}\sum_{i \neq j}\frac{e^2}{4\pi\epsilon_0}\frac{1}{|\mathbf{r}_i - \mathbf{r}_j|} + \frac{1}{2}\sum_{I \neq J}\frac{e^2}{4\pi\epsilon_0}\frac{Z_I Z_J}{|\mathbf{R}_I - \mathbf{R}_J|}.$$

Here, M_I, $\mathbf{R}_I$ and Z_I represent respectively the mass, the position, and the atomic number of the I-th nucleus. Additionally, $\mathbf{r}_i$ represents the positon of the i-th electron. Since a nucleus is many times heavier than an electron, we can utilise the Born-Oppenheimer approximation, treating the nuclei as classical point charges [2]. Also dropping the natural constants, the approximated Hamiltonian then becomes:

$$H_e = -\sum_i \frac{\nabla_i^2}{2} - \sum_{i,I}\frac{Z_I}{|\mathbf{r}_i - \mathbf{R}_I|} + \frac{1}{2}\sum_{i \neq j}\frac{1}{|\mathbf{r}_i - \mathbf{r}_j|}.$$

Now we can write this Hamiltonian in the second quantization notation. An in-depth explanation of the quantisation formalism is out of scope for this research, but the rough idea of second quantisation is that *fields* are being quantised rather than variables. The second quantised form of the electronic Hamiltonian corresponds to

$$H = \sum_{p,q} h_{pq}a_p^\dagger a_q + \frac{1}{2}\sum_{p,q,r,s} h_{pqrs}a_p^\dagger a_q^\dagger a_r a_s, \tag{2.9}$$

where

$$h_{pq} = \int \mathrm{d}\mathbf{x}\phi_p^*(\mathbf{x})\left(-\frac{\nabla^2}{2} - \sum_I \frac{Z_I}{|\mathbf{r} - \mathbf{R}_I|}\right)\phi_q(\mathbf{x}),$$
$$h_{pqrs} = \int \mathrm{d}\mathbf{x}_1\mathrm{d}\mathbf{x}_2 \frac{\phi_p^*(\mathbf{x}_1)\phi_q^*(\mathbf{x}_2)\phi_r(\mathbf{x}_2)\phi_s(\mathbf{x}_1)}{|\mathbf{r}_1 - \mathbf{r}_2|}.$$

and $a_p^\dagger$ and a_q are respectively the fermionic creation and annihilation operators. These operators excite or de-excite electrons into spin-orbitals, and obey the anti-commutation relations

$$\{a_p, a_q^\dagger\} = a_p a_q^\dagger + a_q^\dagger a_p = \delta_{pq},$$
$$\{a_p, a_q\} = \{a_p^\dagger, a_q^\dagger\} = 0.$$

To simulate the electronic structure problem in its second quantized form, a map from operators acting on indistinguishable fermions to operators acting on distinguishable qubits is needed [2]. There are several methods of implementing this map, such as the Jordan-Wigner transformation and the Bravyi-Kitaev encoding.
The Jordan-Wigner transformation writes the annihalation and creation operators in terms of Pauli matrices, such that [11]

$$a_j \rightarrow I^{\otimes n-j-1} \otimes \sigma_+ \otimes \sigma_z^{\otimes j} \text{ and } a_j^\dagger \rightarrow I^{\otimes n-j-1} \otimes \sigma_- \otimes \sigma_z^{\otimes j}$$

where

$$\sigma_+ = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}, \ \sigma_- = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}.$$

The Bravyi-Kitaev expression for the operators in terms of Pauli matrices is significantly more involved than the respective Jordan-Wigner expression, and can be found in [12]. Applying the Jordan-Wigner encoding to the second quantised Hamiltonian of the hydrogen molecule will result in the following representation [12]:

$$\begin{aligned}\hat{H}_{BK} =& -0.81261I + 0.171201\sigma_0^z + 0.16862325\sigma_1^z - 0.2227965\sigma_2^z + 0.171201\sigma_1^z\sigma_0^z \\ &+ 0.12054625\sigma_2^z\sigma_0^z + 0.17434925\sigma_3^z\sigma_1^z + 0.04532175\sigma_2^x\sigma_1^z\sigma_0^x + 0.04532175\sigma_2^y\sigma_1^z\sigma_0^y \\ &+ 0.165868\sigma_2^z\sigma_1^z\sigma_0^z + 0.12054625\sigma_3^z\sigma_2^z\sigma_0^z - 0.2227965\sigma_3^z\sigma_2^z\sigma_1^z \\ &+ 0.04532175\sigma_3^z\sigma_2^x\sigma_1^z\sigma_0^x + 0.04532175\sigma_3^z\sigma_2^y\sigma_1^z\sigma_0^y + 0.165868\sigma_3^z\sigma_2^z\sigma_1^z\sigma_0^z.\end{aligned} \tag{2.10}$$

This Hamiltonian is well suited to quantum computers as it can be encoded in one 8×8 matrix acting on a four-qubit state.

2.2.1   State preparation and the quantum Zeno effect

Any quantum algorithm for ground state energy estimation, such as the one discussed in this thesis, requires (eigen)state preparation, as $H_k \left| \psi \right\rangle = E_k \left| \psi \right\rangle$ only holds when H is acting on an eigenstate. In the spectrum by quantum walk algorithm, phase estimation, the process of retrieving the phase of a unitary operator (discussed in greater detail in section 3.4) performed as a projective energy measurement [1]. Then, it follows that for some approximation of the eigenstate $\left| \tilde{\phi} \right\rangle$, the state of the system collapses to the actual ground state $\left| \phi \right\rangle$ with probability $\left| \langle \tilde{|\phi\rangle} \phi \rangle \right|^2$. Depending on the form of the Hamiltonian, obtaining $\left| \tilde{\phi} \right\rangle$ can be a difficult task. Poulin et al. [1] offer a method for obtaining this approximation, which is almost equivalent to adiabatic state preparation, but has the advantage that only the time-independent Hamiltonian is evolved in time, opposed to the time-dependent implementation of adiabatic state preparation.

In this state preparation sequence, we first define the Hamiltonian to be a composite sum of a simpler Hamiltonian, H_0, and some other interaction term V. This simpler Hamiltonian H_0 is one of which the ground state is known. We can then rewrite the Hamiltonian as a function of some parameter g, where V is linearly related to g:

$$H = H_0 + gV$$

We prepare the physical register of our quantum computer in the simpler ground state of H_0, which in our notation equals the Hamiltonian for $g = 0$. Then we perform a sequence of energy measurements. In each iteration, the starting state is $\left| \phi(g) \right\rangle$. Then, the algorithm, described in later sections, is applied for the Hamiltonian with a g value of $g + dg$. The resulting state of this measurement is $\left| \phi(g + dg) \right\rangle$. The reason this method is useful, is that for a small enough dg, the probability $\left| \langle \phi(g) | \phi(g + dg) \rangle \right|^2$ that measuring the energy for the Hamiltonian $H(g + dg)$ on the state $\left| \phi(g) \right\rangle$ results in the ground energy is close to 1 [1]. This relation is governed by the quantum Zeno effect, in which frequent measurements blocks time evolution of the system and freeze the state to an eigenstate depending on the measurement basis [13]. Thus, given an initialized register in the ground state of H_0, the measurement sequence described in the following sections will result in the desired ground state of the true Hamiltonian.

Chapter 3

Spectrum by Quantum Walk

In this chapter, the Spectrum by Quantum Walk algorithm, proposed by Poulin et al. [1], is discussed.

make bigger

3.1 Operators of the Algorithm

The starting point of this algorithm is the Hamiltonian of the system aimed to simulate:

$$H = \sum_{j=0}^{N} \alpha_j P_j.$$

This Hamiltonian is sum of N multi-qubit Pauli operator multiplied by some real coefficient. Unlike the sum of the second-quantized Hamiltonian in equation 2.9, the number of terms here does not have a direct physical meaning, but is rather dependent on the Jordan-Wigner, Bravyi-Kitaev or parity encoding. A multi-qubit Pauli operator is a tensor product of the four Pauli operators I, X, Y, Z.

Before we move forward, we must ensure that the Hamiltonian we are working with is normalized, as we will map the coefficients to a quantum register in the algorithm. We do so by introducing a scaling factor of $\mathcal{N} = \sum_{j=0}^{N} |\alpha_j|$ [1]. We absorb the sign of the coefficients into P_j.

Then our new, rescaled Hamiltonian, which we will call $\bar{H}$, takes the following form:

$$\bar{H} = \frac{H}{\mathcal{N}} = \sum_{j} |\beta_j|^2 P_j, \tag{3.1}$$

where $\beta_j = \sqrt{|\alpha_j|/\mathcal{N}}$. Due to the nature of the scaling factor $\mathcal{N}$, $\sum_{j} |\beta_j|^2 = 1$. The eigenstates and eigenvalues of the rescaled Hamiltonian are

$$\bar{H} |\phi_k\rangle = \bar{E}_k |\phi_k\rangle. \tag{3.2}$$

The Hamiltonian acts on the n qubits required to store the wavefunction of the system and its matrix representation has dimension $2^n \times 2^n$. It therefore also has 2^n eigenvalue/eigenstate pairs. Since we do not know the exact value of the eigenvalue or eigenstate, we generalize by referring to some eigenstate with the subscript k, where $k = 0$ refers to the ground state.

Later we will define the operator W. This operator will act on two registers, the `ancillary` register and the `physical` register. We shall soon define certain unitary operations of the form $A \otimes B$, in which A will act on the `ancillary` register and B on the `physical` register. Following Poulin et al. [1], we define $|\beta\rangle$ to be the state that results from applying B to the zero-state.

$$|\beta\rangle = B |0\rangle = \sum_{j} \beta_j |j\rangle,$$

where $\sum_j \beta_j$ are the coefficients from equation 3.1. Later, we will initialize the `ancillary` register in the state $|\beta\rangle$. Remember that β_j are the coefficients of the rescaled Hamiltonian, so the bit string basis states $|j\rangle$ are in one-to-one correspondence with the coefficients of the Hamiltonian. As the sum of β_j is normalized, the resulting state $|\beta\rangle$ is in superposition of the N basis states, depending on the coefficients of the multi-qubit Pauli operators of the Hamiltonian. Next, we define the tranformations S and V as

$$S = (B(I - 2\,|0\rangle\,\langle 0|)B^\dagger) \otimes I = (I - 2|\beta\rangle\langle\beta|) \otimes I, \tag{3.3}$$

$$V = \sum_j |j\rangle\,\langle j| \otimes P_j. \tag{3.4}$$

These operations are unitary, which we will prove now.

$$
\begin{aligned}
SS^\dagger &= ((I - 2|\beta\rangle\langle\beta|) \otimes I)^\dagger((I - 2|\beta\rangle\langle\beta|) \otimes I) \\
&= (I - 2|\beta\rangle\langle\beta|)^\dagger(I - 2|\beta\rangle\langle\beta|) \otimes I^2 \\
&= (I - 2(|\beta\rangle\langle\beta|)^\dagger)(I - 2|\beta\rangle\langle\beta|)
\end{aligned}
$$

As $(AB)^\dagger = B^\dagger A^\dagger$,

$$
\begin{aligned}
&= (I - 2\,\langle\beta|^\dagger\,|\beta\rangle^\dagger)(I - 2|\beta\rangle\langle\beta|)\,textas(AB)^\dagger = B^\dagger A^\dagger. \\
&= (I - 2|\beta\rangle\langle\beta|)(I - 2|\beta\rangle\langle\beta|) \\
&= I - 4|\beta\rangle\langle\beta| + 4\,|\beta\rangle\,\langle\beta|\beta\rangle\,\langle\beta| \\
&= I - 4|\beta\rangle\langle\beta| + 4\,|\beta\rangle\sum_l B_l^\dagger \sum_j B_j\,\langle l|j\rangle\,\langle\beta|
\end{aligned}
\tag{3.5}
$$

mathintotext

We will take this opportunity to introduce an identity for the inner product of β and bit-string states. The following holds true for β:

$$\langle\beta|\beta\rangle = \sum_l B_l^\dagger \sum_j B_j\,\langle l|j\rangle = \sum_{l,j} B_l^\dagger \sum_j B_j\delta_{lj} = 1.$$

Such inner products will appear many times in the following sections. Continuing the proof of unitarity of S,

$$SS^\dagger = I - 4|\beta\rangle\langle\beta| + 4\,|\beta\rangle\,\langle\beta| = I.$$

The unitarity of V is proven similarly:

$$
\begin{aligned}
VV^\dagger &= (\sum_j |j\rangle\langle j| \otimes P_j)^\dagger \cdot \sum_j |j\rangle\langle j| \otimes P_k \\
&= \sum_j |j\rangle\langle j| \otimes P_j \cdot \sum_j |j\rangle\langle j| \otimes P_j\,\text{as}\,P_j^\dagger = P_j. \\
&= \sum_j |j\rangle\langle j| \otimes P_j^2
\end{aligned}
\tag{3.6}
$$

$$\text{As}\ \sum_j^n |j\rangle\langle j| = I \text{ and } \sum_j P_j^2 = I,$$

$$VV^\dagger = I$$

3.2   Orthonormal bases

Poulin et al. [1] postulate the following orthonormal basis, in which S and V preserve the subspace spanned by that basis, which will be the same substate that W will be initialized to:

$$
\left| \varphi_k^0 \right\rangle = \sum_j \beta_j \left| j \right\rangle \otimes \left| \phi_k \right\rangle ,
$$
$$
\left| \varphi_k^1 \right\rangle = \frac{1}{\sqrt{1 - E_k^2}} (V - E_k) \left| \varphi_k^0 \right\rangle .
\tag{3.7}
$$

Here, E_k is the energy corresponding to the k-th eigenstate of the Hamiltonian, $\left| \phi_k \right\rangle$. In this section, we prove that above basis states are normalized and orthogonal. In these proofs, inner products of the Hamiltonian eigenstates will appear. As these eigenstates are orthogonal, the inner product becomes:

$$
\langle \phi_{k'} | \phi_k \rangle = \delta_{k'k}.
\tag{3.8}
$$

In this thesis, we assume that $k' = k$, unless specified otherwise. First, we show that $\left| \varphi_k^0 \right\rangle$ is normalized by taking the inner product with itself:

$$
\begin{aligned}
\left\langle \varphi_k^0 | \varphi_k^0 \right\rangle &= ((\langle \beta | \otimes \langle \phi_k |)(|\beta \rangle \otimes |\phi_k \rangle)) \\
&= \langle \beta | \beta \rangle \otimes \langle \phi_k | \phi_k \rangle \\
&= 1.
\end{aligned}
\tag{3.9}
$$

Similarly, we can show that $\left| \varphi_k^1 \right\rangle$ is normalized:

$$
\begin{aligned}
\left\langle \varphi_k^1 | \varphi_k^1 \right\rangle &= \left\langle \varphi_k^0 \right| \frac{1}{\sqrt{1 - E_k^2}} (V - E_k) \frac{1}{\sqrt{1 - E_k^2}} (V - E_k) \left| \varphi_k^0 \right\rangle \\
&= \left\langle \varphi_k^0 \right| \frac{1}{1 - E_k^2} (V - E_k)^2 \left| \varphi_k^0 \right\rangle \\
&= \frac{1}{1 - E_k^2} \left\langle \varphi_k^0 \right| (V^2 - 2 E_k V + E_k^2) \left| \varphi_k^0 \right\rangle \\
&= \frac{1}{1 - E_k^2} \left\langle \varphi_k^0 \right| (I - 2 E_k V + E_k^2) \left| \varphi_k^0 \right\rangle
\end{aligned}
\tag{3.10}
$$

One term in this expression is the expectation value of V in the $\left| \varphi_k^0 \right\rangle$-basis. As finding the expectation value is slightly involved, we first show the proof of the expectation value before continuing with the normalization proof:

$$
\begin{aligned}
\left\langle \varphi_k^0 \right| V \left| \varphi_k^0 \right\rangle &= \left\langle \varphi_k^0 \right| \left(\sum_j |j\rangle\langle j| \otimes P_j \right) \sum_l \beta_l |l\rangle \otimes |\phi_k\rangle \\
&= \left\langle \varphi_k^0 \right| \sum_j \sum_l \beta_l |j\rangle \langle j|l\rangle \otimes P_j |\phi_k\rangle \\
&= \left(\sum_m \beta_m^* \langle m| \otimes \langle \phi_k| \right) \sum_j \sum_l \beta_l |j\rangle \delta_{j,l} \otimes P_j |\phi_k\rangle \\
&= \sum_m \sum_j \beta_m^* \beta_j \langle m|j\rangle \otimes \langle \phi_k | P_j \phi_k \rangle \\
&= \sum_m \sum_j \beta_m^* \beta_j \delta_{m,j} \otimes \langle \phi_k | P_j \phi_k \rangle \\
&= \sum_j |\beta_j|^2 \langle \phi_k | P_j \phi_k \rangle \\
&= \sum_j \left\langle \phi_k \middle| |\beta_j|^2 P_j \phi_k \right\rangle .
\end{aligned}
\tag{3.11}
$$

As $\bar{H}\ket{\phi_k} = E_k\ket{\phi_k}$ and $\bar{H} = \sum_j |\beta_j|^2 P_j$, we have that:

$$\bra{\varphi_k^0} V \ket{\varphi_k^0} = \braket{\phi_k | E_k \phi_k} = E_k. \tag{3.12}$$

We can now continue the proof of orthonormality of $\ket{\varphi_k^1}$ where we left off, using equation 3.12. The expression becomes:

$$
\begin{aligned}
\braket{\varphi_k^1 | \varphi_k^1} &= \frac{1}{1 - E_k^2} \bra{\varphi_k^0} (I - 2E_k V + E_k^2) \ket{\varphi_k^0} \\
&= \frac{1}{1 - E_k^2} \bra{\varphi_k^0} (I - 2E_k^2 + E_k^2) \ket{\varphi_k^0} \\
&= \frac{1}{1 - E_k^2} \bra{\varphi_k^0} (I - E_k^2) \ket{\varphi_k^0} \\
&= \frac{1 - E_k^2}{1 - E_k^2} \braket{\varphi_k^0 | \varphi_k^0} \\
&= 1.
\end{aligned}
\tag{3.13}
$$

Finally, we prove that the two basis states are orthogonal, using equation 3.12:

$$
\begin{aligned}
\braket{\varphi_k^0 | \varphi_k^1} &= \bra{\varphi_k^0} \frac{1}{\sqrt{1 - E_k^2}} (V - E_k) \ket{\varphi_k^0} \\
&= \bra{\varphi_k^0} \frac{1}{\sqrt{1 - E_k^2}} (E_k - E_k) \ket{\varphi_k^0} \\
&= 0.
\end{aligned}
\tag{3.14}
$$

3.2.1 Blockdiagonality

The unitary property of S and V allow them to be put in block diagonal form with 2×2 blocks [1]. A general block diagonal matrix only has nonzero entries in the blocks of size $n \times n$ on the diagonal, which for $n = 2$ looks like the following:

$$
\begin{pmatrix}
\begin{array}{cc|cc}
a & b & 0 & 0 \\
c & d & 0 & 0 \\
\hline
0 & 0 & e & f \\
0 & 0 & g & h
\end{array}
\end{pmatrix}.
$$

We can put the expectation values of S and V in one of such matrices in the space of $\ket{\varphi_k^0}$ and $\ket{\varphi_k^1}$. Putting the operators S and V into block diagonal form with placeholder values in aforementioned space creates the following structure:

$$
\begin{array}{c|cc|cc|c|cc}
 & \ket{\varphi_0^0} & \ket{\varphi_0^1} & \ket{\varphi_1^0} & \ket{\varphi_1^1} & \cdots & \ket{\varphi_k^0} & \ket{\varphi_k^1} \\
\hline
\bra{\varphi_0^0} & V_{11} & V_{21} & 0 & 0 & & 0 & 0 \\
\bra{\varphi_0^1} & V_{12} & V_{22} & 0 & 0 & & 0 & 0 \\
\hline
\bra{\varphi_1^0} & 0 & 0 & V_{33} & V_{43} & & 0 & 0 \\
\bra{\varphi_1^1} & 0 & 0 & V_{43} & V_{44} & & 0 & 0 \\
\hline
\vdots & & & & & \ddots & & \\
\hline
\bra{\varphi_k^0} & 0 & 0 & 0 & 0 & & V_{ii} & V_{ij} \\
\bra{\varphi_k^1} & 0 & 0 & 0 & 0 & & V_{ji} & V_{jj}
\end{array}
\tag{3.15}
$$

The off-diagonal elements are zero as $\braket{\phi_k | \phi_l} = \delta_{k,l}$. As the expectation values of S and V in the basis defined in 3.7 will be useful later in this thesis, we shall now try to find these values. Before we do so, it will be convenient to first find the result of S applied to $\ket{\varphi_k^0}$:

$$S \left|\varphi_k^0\right\rangle = ((I - 2 \left|\beta\right\rangle\!\left\langle\beta\right|) \otimes I) \left|\beta\right\rangle \otimes \left|\phi_k\right\rangle$$
$$= \left|\beta\right\rangle \otimes \left|\phi_k\right\rangle - 2 \left|\beta\right\rangle \otimes \left|\phi_k\right\rangle \tag{3.16}$$
$$= - \left|\varphi_k^0\right\rangle.$$

Then the expectation value of S in $\left|\varphi_0^0\right\rangle$ simply becomes:

$$\left\langle\varphi_0^0\right| S \left|\varphi_0^0\right\rangle = \left\langle\varphi_0^0\right| \left(- \left|\varphi_k^0\right\rangle\right)$$
$$= - \left\langle\varphi_0^0\middle|\varphi_0^0\right\rangle \tag{3.17}$$
$$= -1.$$

Next, we find the expectation value of the offdiagonal entries in the matrix:

$$\left\langle\varphi_0^0\right| S \left|\varphi_0^1\right\rangle = \left\langle\varphi_0^0\right| \left(- \left|\varphi_0^1\right\rangle\right) \text{ using 3.16}$$
$$= - \left\langle\varphi_0^0\middle|\varphi_0^1\right\rangle \tag{3.18}$$
$$= 0.$$

Because the operator S is symmetric around the diagonal, it is self adjoint and from the identity [9]
$$\left\langle\psi\right| A^\dagger \left|\phi\right\rangle = \left\langle\phi\right| A \left|\psi\right\rangle^*$$
we can conclude that $\left\langle\varphi_0^0\right| S \left|\varphi_0^1\right\rangle = \left\langle\varphi_0^1\right| S \left|\varphi_0^0\right\rangle = 0$.

$$\left\langle\varphi_0^1\right| S \left|\varphi_0^1\right\rangle = \left\langle\varphi_0^1\right| S(V - E_k) \left|\varphi_0^0\right\rangle$$
$$= \left\langle\varphi_0^1\right| SV - SE_0 \left|\varphi_0^0\right\rangle)$$
$$= (\left\langle\varphi_0^1\right| SV \left|\varphi_0^0\right\rangle + E_k \left\langle\varphi_0^1\right| S \left|\varphi_0^0\right\rangle)$$
$$= \left\langle\varphi_0^1\right| SV \left|\varphi_0^0\right\rangle$$
$$= \left\langle\varphi_0^1\right| (I - 2\left|\beta\right\rangle\!\left\langle\beta\right|)V \left|\varphi_0^0\right\rangle$$
$$= (\left\langle\varphi_0^1\right| V \left|\varphi_0^0\right\rangle - 2 \left\langle\varphi_0^1\right| \left|\beta\right\rangle\!\left\langle\beta\right|V \left|\varphi_0^0\right\rangle)$$
$$= \frac{\sqrt{1 - E_0^2}}{\sqrt{1 - E_0^2}} - 2 \left\langle\varphi_0^1\right| \sum_j \left|\beta_j\right|^2 \left|\beta\right\rangle \otimes P_j \left|\phi_0\right\rangle \tag{3.19}$$
$$= 1 - \frac{2}{\sqrt{1 - E_0^2}} \left\langle\varphi_0^1\right| (\left|\beta\right\rangle \otimes H \left|\phi_0\right\rangle)$$
$$= 1 - \frac{2E_k}{\sqrt{1 - E_0^2}} \left\langle\varphi_0^1\right| (\left|\beta\right\rangle \otimes \left|\phi_0\right\rangle)$$
$$= 1 - \frac{2E_k}{\sqrt{1 - E_0^2}} \left\langle\varphi_0^1\middle|\varphi_0^0\right\rangle$$
$$= 1.$$

The block-diagonal matrix of S is now completed, and has the following form:

$$S = \begin{array}{c} \\ \left\langle\varphi_0^0\right| \\ \left\langle\varphi_0^1\right| \end{array} \begin{array}{cc} \left|\varphi_0^0\right\rangle & \left|\varphi_0^1\right\rangle \\ \left[\begin{array}{cc} -1 & 0 \\ 0 & 1 \end{array} \right]. \end{array} \tag{3.20}$$

Since there is no k-dependency, this matrix can be generalised for all k-values, which gives the following:

$$S = \begin{array}{c} \\ \langle \varphi_k^0 | \\ \langle \varphi_k^1 | \end{array} \begin{array}{cc} |\varphi_k^0\rangle & |\varphi_k^1\rangle \\ \left[\begin{array}{cc} -1 & 0 \\ 0 & 1 \end{array} \right]. \end{array} \tag{3.21}$$

Let us find the equivalent matrix for the operator V. In equation 3.12, it has been shown that $\langle \varphi_k^0 | V | \varphi_k^0 \rangle = E_k$. Next, we find the off-diagonal elements:

$$
\begin{aligned}
\langle \varphi_0^0 | V | \varphi_0^1 \rangle &= \langle \varphi_0^0 | \frac{1}{\sqrt{1 - E_0^2}} (V^2 - E_0 V) | \varphi_0^0 \rangle \\
&= \frac{1}{\sqrt{1 - E_0^2}} \left(\langle \varphi_0^0 | \varphi_0^0 \rangle - E_0 \langle \varphi_0^0 | V | \varphi_0^0 \rangle \right) \\
&= \left(\langle \varphi_0^0 | \varphi_0^0 \rangle - E_0^2 \langle \phi_0 | \phi_0 \rangle \right) \text{ using 3.12} \\
&= \sqrt{1 - E_0^2}.
\end{aligned}
\tag{3.22}
$$

As V is also symmetric and therefore self-adjoint,

$$\langle \psi | A^\dagger | \phi \rangle = \langle \phi | A | \psi \rangle^*$$

holds again and thus

$$\langle \varphi_0^0 | V | \varphi_0^1 \rangle = \langle \varphi_0^1 | V | \varphi_0^0 \rangle = \sqrt{1 - E_0^2}.$$

Lastly, we need to find $\langle \varphi_0^1 | V | \varphi_0^1 \rangle$:

$$
\begin{aligned}
\langle \varphi_0^1 | V | \varphi_0^1 \rangle &= \langle \varphi_0^1 | V \frac{V - E_0}{\sqrt{1 - E_0^2}} | \varphi_0^0 \rangle \\
&= \langle \varphi_0^0 | \frac{1}{1 - E_0^2} (V - E_0)(1 - E_0 V) | \varphi_0^0 \rangle \\
&= \langle \varphi_0^0 | \frac{1}{1 - E_0^2} (V - E_0 - E_0 + E_0^2 V) | \varphi_0^0 \rangle \text{ using equation 3.12} \\
&= \langle \phi_0 | \frac{1}{1 - E_0^2} (-E_0 + E_0^3) | \phi_0 \rangle \\
&= \langle \phi_0 | \left(-E_0 \frac{1 - E_0^2}{1 - E_0^2} | \phi_0 \rangle \right) \\
&= \langle \phi_0 | (-E_0 | \phi_0 \rangle) \\
&= -E_0.
\end{aligned}
\tag{3.23}
$$

Once again, we can compile these values in the block matrix and generalize to any k-value in the range of energy states to obtain:

$$V = \begin{array}{c} \\ \langle \varphi_k^0 | \\ \langle \varphi_k^1 | \end{array} \begin{array}{cc} |\varphi_k^0\rangle & |\varphi_k^1\rangle \\ \left[\begin{array}{cc} E_k & \sqrt{1 - E_0^2} \\ \sqrt{1 - E_0^2} & -E_k \end{array} \right] \end{array} \tag{3.24}$$

3.3   Creating the Unitary Walk Operator

So far, we have defined three unitary operations (B, S and V) and have defined an orthogonal basis $\{|\varphi_k^0\rangle, |\varphi_k^1\rangle\}$. B is the ancillary state preparation operator, creating a superposition of states while incorporating the Hamiltonian coefficients. S is a reflection operator and V is a controlled operator that applies Pauli operators. Poulin et al. [1] define a unitary operator W which preserves the subspace spanned by the orthogonal basis just specified:

$$W = SV. \tag{3.25}$$

As we have defined both S and V in the $\{|\varphi_k^0\rangle, |\varphi_k^1\rangle\}$-basis, we can find the block-diagonal matrix for W as well:

$$
\begin{aligned}
W = SV &= \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} E_k & \sqrt{1-E_k^2} \\ \sqrt{1-E_k^2} & -E_k \end{pmatrix} \\
&= \begin{pmatrix} -E_k & -\sqrt{1-E_k^2} \\ \sqrt{1-E_k^2} & -E_k \end{pmatrix}.
\end{aligned} \tag{3.26}
$$

The operator W has eigenstates $|\varphi_k^\pm\rangle = |\varphi_k^0\rangle \pm i |\varphi_k^1\rangle$ [1]. Let us find the eigenvalues by applying W to its eigenstates:

$$W |\varphi_k^+\rangle = \frac{1}{\sqrt{2}} \left(W |\varphi_k^0\rangle + W i |\varphi_k^1\rangle \right) \tag{3.27}$$

$$= \frac{1}{\sqrt{2}} \left(SV |\varphi_k^0\rangle + i SV \frac{V - E_k}{\sqrt{1-E_k^2}} |\varphi_k^0\rangle \right) \tag{3.28}$$

$$= \frac{1}{\sqrt{2}} \left(SV |\varphi_k^0\rangle + \frac{iS}{\sqrt{1-E_k^2}} I |\varphi_k^0\rangle - \frac{iE_k}{\sqrt{1-E_k^2}} SV |\varphi_k^0\rangle \right) \tag{3.29}$$

$$= \frac{1}{\sqrt{2}} \left(\frac{\sqrt{1-E_k^2} S - i E_k S}{\sqrt{1-E_k^2}} V |\varphi_k^0\rangle + \frac{iS}{\sqrt{1-E_k^2}} |\varphi_k^0\rangle \right). \tag{3.30}$$

We want to obtain a factor of $|\varphi_k^1\rangle$. To do so, we add and subtract a factor of

$$E_k \left(\frac{\sqrt{1-E_k^2} S - i E_k S}{\sqrt{1-E_k^2}} |\varphi_k^0\rangle \right)$$

to obtain the desired $(V - E_k)$ factor in the left term of the equation above. Then, the expression becomes:

$$W |\varphi_k^+\rangle = \frac{1}{\sqrt{2}} \left(\frac{\sqrt{1-E_k^2} S - i E_k S}{\sqrt{1-E_k^2}} (V - E_k) |\varphi_k^0\rangle + \frac{iS}{\sqrt{1-E_k^2}} |\varphi_k^0\rangle + E_k \frac{\sqrt{1-E_k^2} - i E_k}{\sqrt{1-E_k^2}} S |\varphi_k^0\rangle \right) \tag{3.31}$$

Given $\frac{V - E_k}{\sqrt{1-E_k^2}} |\varphi_k^0\rangle = |\varphi_k^1\rangle$ we obtain the following

$$= \frac{1}{\sqrt{2}} \left(\left(\sqrt{1-E_k^2} - i E_k \right) S |\varphi_k^1\rangle + \frac{i + E_k(\sqrt{1-E_k^2} - i E_k)}{\sqrt{1-E_k^2}} S |\varphi_k^0\rangle \right). \tag{3.32}$$

Since $S\left|\varphi_k^1\right\rangle = \left|\varphi_k^1\right\rangle$ and $S\left|\varphi_k^0\right\rangle = -\left|\varphi_k^0\right\rangle$, our expression becomes

$$= \frac{1}{\sqrt{2}}\left(\left(\sqrt{1-E_k^2}-iE_k\right)\left|\varphi_k^1\right\rangle - \left(E_k\frac{\sqrt{1-E_k^2}}{\sqrt{1-E_k^2}}+i\frac{1-E_k^2}{\sqrt{1-E_k^2}}\right)\left|\varphi_k^0\right\rangle\right) \tag{3.33}$$

$$= \frac{1}{\sqrt{2}}\left(\left(\sqrt{1-E_k^2}-iE_k\right)\left|\varphi_k^1\right\rangle - \left(E_k+i\sqrt{1-E_k^2}\right)\left|\varphi_k^0\right\rangle\right) \tag{3.34}$$

$$= \frac{1}{\sqrt{2}}\left(\sqrt{1-E_k^2}-iE_k\right)\left|\varphi_k^1\right\rangle + \frac{1}{\sqrt{2}}\left(-E_k-i\sqrt{1-E_k^2}\right)\left|\varphi_k^0\right\rangle. \tag{3.35}$$

Following Steiger [10], we can factor the coefficients of the basis state into $-E_k-i\sqrt{1-E_k^2}$ and obtain

$$= \left(-E_k-i\sqrt{1-E_k2}\right)\frac{1}{\sqrt{2}}(\left|\varphi_k^0\right\rangle + i\left|\varphi_k^1\right\rangle) \tag{3.36}$$

$$= \left(-E_k-i\sqrt{1-E_k2}\right)\left|\varphi_k^+\right\rangle. \tag{3.37}$$

As $E_k = \cos(\theta_k)$ (which we show in section 3.4) and $|\sin(\theta_k)| = \sqrt{1-\cos^2(\theta_k)}$, we can write this as

$$= (-\cos(\theta_k)-i\sin(\theta_k))\left|\varphi_k^+\right\rangle \tag{3.38}$$

$$= e^{i(\pi+\theta_k)}\left|\varphi_k^+\right\rangle. \tag{3.39}$$

Similarly, we can find that $W\left|\varphi_k^-\right\rangle = e^{i(\pi-\theta_k)}\left|\varphi_k^-\right\rangle$, which implies that the eigenvalues of W are $e^{i(\pi\pm\theta_k)}$.

3.4   Quantum Phase Estimation

In this section, we briefly outline the canonical quantum Phase Estimation Algorithm (PEA) as introduced by Nielsen and Chuang [9]. The norm of eigenvalues of all unitary operators, such as the unitary operator W, is one and the eigenvalues can be written in the form $\exp\{2\pi i\theta\}$. This algorithm prepares the physical register in the $\{\left|\varphi_k^0\right\rangle, \left|\varphi_k^1\right\rangle\}$ subspace, as the set of eigenvalues of W in this specific subspace is a function of the Hamiltonian [1]. As the eigenvalues of W are $e^{i(\pi\pm\theta_k)}$, we can conclude that the eigenstate of the Hamiltonian, which is E_k, is encoded in the phase of W. The exact relation between the two spectra is given by Babbush et al. [14], who describe the same method of spectral encoding of the Hamiltonian

$$\text{spectrum}(H) = \lambda\cos(\arg(\text{spectrum}(\mathcal{W}))). \tag{3.40}$$

Here, arg represents the argument function of complex analysis, and λ is the rescaling factor, equal to $\mathcal{N}$ introduced in equation 3.1. We follow the convention introduced by Poulin et al. [1], where E_k refers to the rescaled energy. Consequently, we can rewrite equation 3.40 as

$$\cos(\theta_k) = E_k. \tag{3.41}$$

We can thus perform an energy measurement by performing phase estimation on W.
Phase estimation makes use of two registers: an ancillary register of size t and a system register, prepared in the eigenstate $|u\rangle$. We assume that the phase of the eigenvalue $\exp\{2\pi i\varphi\}$ can be written as a $0.\varphi_1\varphi_2...\varphi_t$ in binary notation, where t is the size of the ancillary register [9]. First, the qubits of the ancillary register is brought to an equal superposition through a Hadamard transform. Then, a series of controlled-U operations are performed on the system register, where U is raised to successive powers of two. The schematic circuit of this procedure can be seen in figure 3.1.
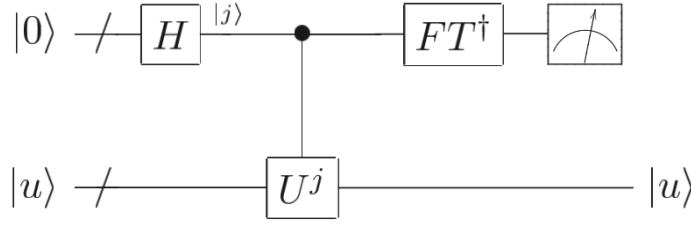The state of the ancillary register after this procedure is

Figure 3.1: Schematic of the phase estimation procedure [9, p. 223].

$$\frac{1}{2^{t/2}} \sum_{k=0}^{2^t-1} e^{2\pi i \varphi k} |k\rangle, \tag{3.42}$$

where φ is the phase of the eigenvalue of U. This first part of the phase estimation algorithm iteratively projects the phase of the unitary operator U onto the state of the ancillary register. In order to retrieve the phase from the ancillary state, we must use the Fourier transform. The quantum Fourier transform is defined as [9]

$$|j\rangle \longrightarrow \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2\pi i j k/N} |k\rangle, \tag{3.43}$$

where $|j\rangle$ is a bit-string state and the sum is over an orthonormal basis $|0\rangle, ..., |N-1\rangle$. From above definition and t is clear that the ancillary state is the Fourier transform of the phase $|\varphi_k\rangle$. The next step of the phase estimation algorithm is to apply the inverse Fourier Transform ($FT^\dagger$ to the ancillary state:

$$\frac{1}{2^{t/2}} \sum_{j=0}^{2^t-1} e^{2\pi i \varphi j} |j\rangle |u\rangle \rightarrow |\tilde{\varphi}\rangle |u\rangle \tag{3.44}$$

By applying a measurement in the computational basis, we now retrieve a good approximation of the phase, $\tilde{\varphi}$. Performing phase estimation on W when the qubits are initialized to $B|0\rangle \otimes \left|\tilde{\phi}_0\right\rangle$ gives $\pi \pm \theta_0$ with probability $\left|\left\langle \phi_0 | \tilde{\phi}_0 \right\rangle\right|^2$. Then we can easily calculate the ground energy, as $E_0 = \cos(\pm\theta_0)$ After the phase estimation process, the qubits are left in the eigenstate of W, $\left|\varphi_k^\pm\right\rangle$. Even though the furnished state is not directly the eigenstate of the Hamiltonian, we can still obtain relevant information of the eigenstate of the Hamiltonian. Since quantum states collapse upon measurement, we cannot simply measure a state once and define the superposition of that state. The general method to obtain information about a circuit or state is by performing repeated measurements of an hermitian observable O. Taking the average over these measurements, we obtain the expectation value of that observable:

$$\bar{O} = \langle \varphi | O | \varphi \rangle.$$

Similarly, we can find the expectation value of some observable in the eigenstate of W and in the next section we prove that this gives us a value for the expectation values of observables in the Hamiltonian eigenstates themselves.

3.5   Hamiltonian eigenstate expectation values

Let σ be some multi-qubit Pauli operator. The expectation value of σ in the eigenstate of W then can be written as the following.

$$
\begin{aligned}
\langle \varphi_k^{\pm} | \, \sigma \, | \varphi_k^{\pm} \rangle &= \left(\frac{1}{\sqrt{2}} \langle \varphi_k^0 | \mp i \langle \varphi_k^1 | \right) \sigma \left(\frac{1}{\sqrt{2}} | \varphi_k^0 \rangle \pm i | \varphi_k^1 \rangle \right) \\
&= \frac{1}{2} \left(\langle \varphi_k^0 | \, \sigma \, | \varphi_k^0 \rangle \pm i \langle \varphi_k^0 | \, \sigma \, | \varphi_k^1 \rangle \mp i \langle \varphi_k^1 | \, \sigma \, | \varphi_k^0 \rangle + \langle \varphi_k^1 | \, \sigma \, | \varphi_k^1 \rangle \right)
\end{aligned}
\tag{3.45}
$$

We now consider each term individually.

$$
\begin{aligned}
\langle \varphi_k^0 | \, \sigma \, | \varphi_k^0 \rangle &= \langle \beta | \otimes \langle \phi_k | \, \sigma \, | \beta \rangle \otimes | \phi_k \rangle \\
&= \langle \beta | \beta \rangle \otimes \langle \phi_k | \, \sigma \, | \phi_k \rangle \\
&= \langle \phi_k | \, \sigma \, | \phi_k \rangle
\end{aligned}
\tag{3.46}
$$

$$
\begin{aligned}
\langle \varphi_k^0 | \, \sigma \, | \varphi_k^1 \rangle &= \frac{1}{\sqrt{1 - E_k^2}} \langle \beta | \otimes \langle \varphi_k | \, \sigma (V - E_k) \, | \beta \rangle \otimes | \phi_k \rangle \\
&= \frac{1}{\sqrt{1 - E_k^2}} \langle \beta | \otimes \langle \varphi_k | \, \sigma \left(\sum_j \beta_j | j \rangle \otimes P_j | \phi_k \rangle - E_k | \beta \rangle \otimes | \phi_k \rangle \right) \\
&= \frac{1}{\sqrt{1 - E_k^2}} \left(\sum_j | \beta_j^2 | \langle \phi_k | \, \sigma P_j \, | \phi_k \rangle - E_k \langle \phi_k | \, \sigma \, | \phi_k \rangle \right) \\
&= \frac{1}{\sqrt{1 - E_k^2}} \left(E_k \langle \phi_k | \, \sigma \, | \phi_k \rangle - E_k \langle \phi_k | \, \sigma \, | \phi_k \rangle \right) \\
&= 0
\end{aligned}
\tag{3.47}
$$

$$
\begin{aligned}
\langle \varphi_k^1 | \, \sigma \, | \varphi_k^0 \rangle &= (\langle \varphi_k^0 | \, \sigma \, | \varphi_k^1 \rangle)^* \\
&= 0
\end{aligned}
\tag{3.48}
$$

$$
\begin{aligned}
\langle \varphi_k^1 | \, \sigma \, | \varphi_k^1 \rangle) &= \frac{1}{1 - E_k^2} \langle \varphi_k^0 | \, (V - E_k) \sigma (V - E_k) \, | \varphi_k^0 \rangle \\
&= \frac{1}{1 - E_k^2} \langle \varphi_k^0 | \, (V \sigma V - V \sigma E_k - E_k \sigma V + E_k \sigma E_k) \, | \varphi_k^0 \rangle \\
&= \frac{1}{1 - E_k^2} \langle \varphi_k^0 | \, V \sigma V \, | \varphi_k^0 \rangle - \langle \varphi_k^0 | \, V \sigma E_k \, | \varphi_k^0 \rangle \\
&\quad - \langle \varphi_k^0 | \, E_k \sigma V \, | \varphi_k^0 \rangle + \langle \varphi_k^0 | \, E_k \sigma E_k \, | \varphi_k^0 \rangle)
\end{aligned}
\tag{3.49}
$$

We once again consider each expectation value individually.

$$
\langle \varphi_k^0 | \, E_k \sigma E_k \, | \varphi_k^0 \rangle = E_k^2 \langle \phi_k | \, \sigma \, | \phi_k \rangle
$$

Using equation 2.7, the following holds.

$$
\begin{aligned}
\langle \varphi_k^0 | \, E_k \sigma V \, | \varphi_k^0 \rangle &= E_k \langle \varphi_k^0 | \, \sigma \, | \beta \rangle \otimes P_j | \phi_k \rangle \\
&= E_k \langle \beta | \beta \rangle \otimes \langle \phi_k | \, \sigma P_j \, | \phi_k \rangle \\
&= E_k \sum_j \langle \phi_k | \, \sigma | \beta_j |^2 P_j \, | \phi_k \rangle \\
&= E_k^2 \langle \phi_k | \, \sigma \, | \phi_k \rangle
\end{aligned}
\tag{3.50}
$$

As $\langle \varphi_k^0 | V\sigma E_k | \varphi_k^0 \rangle$ is symmetric to $\langle \varphi_k^0 | E_k\sigma V | \varphi_k^0 \rangle$, we can conclude that $\langle \varphi_k^0 | V\sigma E_k | \varphi_k^0 \rangle = E_k^2 \langle \phi_k | \sigma | \phi_k \rangle$

$$
\begin{aligned}
\langle \varphi_k^0 | V\sigma V | \varphi_k^0 \rangle &= \langle \beta | \otimes \langle \phi_k | V\sigma V | \beta \rangle \otimes | \phi_k \rangle \\
&= \sum_l \langle l | \beta_l^\dagger \otimes \langle \phi_k | P_l \sigma \sum_j \beta_j | j \rangle \otimes P_j | \phi_k \rangle \\
&= \sum_j |\beta_j|^2 \langle j | j \rangle \otimes \langle \phi_k | P_j \sigma P_j | \phi_k \rangle \\
&= \sum_j |\beta_j|^2 \langle \phi_k | P_j \sigma P_j | \phi_k \rangle
\end{aligned}
\tag{3.51}
$$

We split this sum up into two separate sums, where an extra constraint is placed on the (anti)commutation relation of σ and P_j.

$$
= \sum_{\substack{j \\ [\sigma, P_j = 0]}} |\beta_j|^2 \langle \phi_k | P_j \sigma P_j | \phi_k \rangle + \sum_{\substack{j \\ \{\sigma, P_j = 0\}}} |\beta_j|^2 \langle \phi_k | P_j \sigma P_j | \phi_k \rangle
$$

In the sum where σ and P_j commute, we can switch the order of σ and P_j. We can do the same for the sum where σ and P_j commute, but that results in a sign change, as $AB = -BA$ for $\{A, B\} = 0$. This gives

$$
= \sum_{\substack{j \\ [\sigma, P_j = 0]}} |\beta_j|^2 \langle \phi_k | P_j P_j \sigma | \phi_k \rangle - \sum_{\substack{j \\ \{\sigma, P_j = 0\}}} |\beta_j|^2 \langle \phi_k | P_j P_j \sigma | \phi_k \rangle .
$$

As P_j is unconditionally unitary, $P_j^2 = 1$. We can now merge the two sums back together by defining an additional operator σ :

$$
\sigma \star P_j = 0 \iff [\sigma, P_j] = 0, \quad \sigma \star P_j = 1 \iff \{\sigma, P_j\} = 0.
$$

Then the two sums can be written as

$$
\sum_j |\beta_j|^2 (-1)^{\sigma \star P_j} \langle \phi_k | \sigma | \phi_k \rangle
$$

Following the convention introduced by Poulin et al. [1], we define another variable Γ:

$$
\Gamma_\sigma = \sum_j |\beta_j|^2 (-1)^{\sigma \star P_j}
\tag{3.52}
$$

Then the expectation value can be written as

$$
\langle \varphi_k^0 | V\sigma V | \varphi_k^0 \rangle = \Gamma_\sigma \langle \phi_k | \sigma | \phi_k \rangle
\tag{3.53}
$$

Filling in the individual expecation values in equation 2.17, we obtain

$$
\begin{aligned}
\langle \varphi_k^1 | \sigma | \varphi_k^1 \rangle) &= \frac{1}{1 - E_k^2} (\langle \varphi_k^0 | V\sigma V | \varphi_k^0 \rangle - \langle \varphi_k^0 | V\sigma E_k | \varphi_k^0 \rangle \\
&\quad - \langle \varphi_k^0 | E_k\sigma V | \varphi_k^0 \rangle + \langle \varphi_k^0 | E_k\sigma E_k | \varphi_k^0 \rangle) \\
&= \frac{\Gamma_\sigma - 2E_k^2 + E_k^2}{1 - E_k^2} \langle \phi_k | \sigma | \phi_k \rangle \\
&= \frac{\Gamma_\sigma - E_k^2}{1 - E_k^2} \langle \phi_k | \sigma | \phi_k \rangle .
\end{aligned}
\tag{3.54}
$$

We can now fill this expression back in equation 2.13 and obtain the value for the expectation value of some multi-qubit Pauli operator in the $\varphi_k^{\pm}$-basis.

$$\langle \varphi_k^{\pm} | \sigma | \varphi_k^{\pm} \rangle = \frac{1}{2}(1 + \frac{\Gamma_{\sigma} - E_k^2}{1 - E_k^2}) \langle \phi_k | \sigma | \phi_k \rangle$$

Since Γ_{σ} can be calculated classically, we obtain static expectation values without having to prepare the eigenstate of the Hamiltonian. However, while obtaining these expectation values is very useful, preparing the `physical` register in the eigenstate of the Hamiltonian may also be desired. To do so, we apply $B^{\dagger}$ to the `ancillary` register of the post-measurement state $\left| \varphi_k^{\pm} \right\rangle$, in which $|\varphi_k\rangle$ corresponds to the E_k measured. This gives

$$
\begin{aligned}
\left(B^{\dagger} \otimes I \right) \left| \varphi_k^{\pm} \right\rangle &= B^{\dagger} \left(\frac{1}{\sqrt{2}} \left| \varphi_k^0 \right\rangle \mp i \left| \varphi_k^1 \right\rangle \right) \\
&= \frac{1}{\sqrt{2}} (B^{\dagger} B \left| 0 \right\rangle \otimes \left| \phi_k \right\rangle \pm \left(B^{\dagger} \otimes I \right) i \left| \varphi_k^1 \right\rangle \\
&= \frac{1}{\sqrt{2}} (\left| 0 \right\rangle \otimes \left| \phi_k \right\rangle \pm \left(B^{\dagger} \otimes I \right) i \left| \varphi_k^1 \right\rangle
\end{aligned}
\tag{3.55}
$$

The system can now be in two possible states: either the ancillary register is $|0\rangle$ and the physical register is in the desired state of $|\phi_k\rangle$, or the system is in $\left(B^{\dagger} \otimes I \right) i \left| \varphi_k^1 \right\rangle$. According to Steiger [10], the latter state has no overlap with the ancillary state $|0\rangle$, so the probability that the physical register is in the state $|\phi_k\rangle$ is $\frac{1}{2}$, given the superposition of 3.55. We can then use an additional ancillary qubit to determine whether the ancilla register is in $|0\rangle$ without collapsing the system. We can achieve this by performing the operation described in 4.1, where the target qubit is the additional ancillary qubit, on which we perform a measurement afterwards. If this measurement returns 1, then the Hamiltonian eigenstate has been sucessfully prepared. Otherwise, we can apply B again and perform phase estimation of W to return to $\left| \varphi_k^{\pm} \right\rangle$, where we can retry the process just described. Either way, we can prepare the ground state of the Hamiltonian without collapsing the state of the system.

Chapter 4

Implementation and Protocol

In this chapter, we translate the Spectrum by quantum walk algorithm from theory to a physically realizable implementation, by first providing a protocol with relevant quantum circuits. Then we show an implementation in *ProjectQ* for the quantum chemistry problem of two systems, the element H and the molecule H_2. Note that there is a distinct difference between these two: while they both carry the name hydrogen, H_2 is referred to as *molecular* hydrogen while H is the *molecule* hydrogen.

4.1 Protocol

Throughout the description of the protocol of the algorithm, we consider the specifics of simulating the Hamiltoniam of the hydrogen molecule. The first step is to assign the necessary quantum registers in the quantum computer. There are in total three registers: the `physical` register, the `ancillary` register and the `counting` register. The `physical` register is of size n, where n is the number of qubits that the Hamiltonian acts on. The `ancillary` register is of size $\log(N+1)$, the ceiling of the number of qubits required to describe N terms, where N is the number of terms in the Hamiltonian. The `counting` register is the auxilliary register of the phase estimation procedure of W. The size of the register, denoted as t, depends on the desired success rate and accuracy of the phase estimation.

Next, the registers are to be initialized. We initialize the `counting` register to $|0\rangle^{\otimes t}$ and subsequently apply $H^{\otimes t}$. We aim to prepare the rest of the system in the state

$$B|0\rangle \otimes |\tilde{\phi_0}\rangle$$

The B operator maps the $|0\rangle$-state to $= \sum_j \beta_j |j\rangle$, so B can be represented as an N-dimensional matrix where the leftmost column is filled with the coefficients of the rescaled Hamiltonian. This matrix is not unitary yet but as the rest of the matrix is not constrained, the matrix can be made unitary by the Gram-Schmidt process. There exists a myriad of methods for decomposing this matrix into single-qubit and CNOT gates. In our implementation, a pre-existing state-preparation function is used, which is an implementation of the general state preparation circuit introduced by Mottonen et al. [15]. This method uses a sequence of uniformly controlled rotations with an upper bound for the complexity of $2^{n+2} - 4n - 4$ CNOT gates and $2^{n+2} - 5$ one-qubit operations.

The `physical` register is to be prepared in $\left|\tilde{\phi_0}\right\rangle$, which has a large overlap with the actual ground state $|\phi_0\rangle$. We dicussed in more detail how to obtain such approximation for highly complex systems in 2.2.1. In our implementation of the hydrogen systems, eigenstates can be obtained by diagonalization as the Hamiltonian matrices are not computationally complex. We can artificially create an approximation of the ground state by mapping a superposition of the ground state and some excited states to the physical register. After the qubits are initialized, we need to define the subroutines that compose W. Let us first remind ourselves of the definition of the reflection operator S:

$$S = B^\dagger \left(I - 2 \left| 0 \right\rangle \left\langle 0 \right| \right) B$$

$(I - 2 \left| 0 \right\rangle \left\langle 0 \right|)$ maps $\left| 0 \right\rangle^{\otimes n}$ to $- \left| 0 \right\rangle^{\otimes n}$ and acts as identity on all the other state, reminiscent of an inversely acting N-Controlled-Z operation. We can achieve this subroutine by first applying NOT to the entire register, then performing a N-Controlled-Z operation and ultimately re-applying NOT to the register. A circuit schematic of this subroutine is shown below.
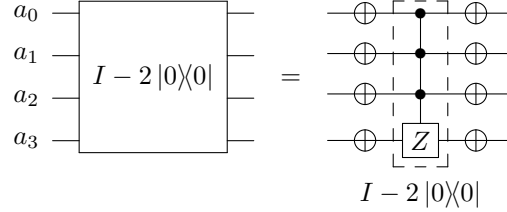
Figure 4.1: Schematic of the subroutine $(I - 2 \left| 0 \right\rangle \left\langle 0 \right|)$ for $n = 4$.

This operation is enclosed between $B^\dagger$ and B. Moving onto V, this operation applies the j-th multi-qubit Pauli operator to the `physical` register, controlled by the `ancillary` register. This means that To do so, we introduce another convention, namely a not gate applied to the target bit, conditional on the control bit being set to zero [9]. This effect is achieved by

$$X_{\text{control}} CNOT_{\text{control, target}} X_{\text{control}} \left| \psi \right\rangle \tag{4.1}$$
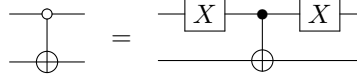
.

Figure 4.2: Circuit schematic of the alternate controlled NOT operation, dependent on the control qubit being one.

This operation allows us to construct a circuit where an operation is applied, conditional on a register being set to a specific bit-string state. For example, the V operation for $j = 5, \left| 5 \right\rangle = \left| 101 \right\rangle$: $\left| 5 \right\rangle \left\langle 5 \right| \otimes P_5$ will only apply P_5 on the `physical` register if the first ancillary qubit is 1, the second qubit 0 and the third qubit 1.

The circuit of the unitary walk operator W is constructed by the successive application of the circuits of S and V. Next, we perform phase estimation, whose circuit schematic can be shown in section 3.2. To recap, phase estimation consists of a series of controlled operations W, where for each qubit in the prepared `counting` register, W is applied to the `physical` and `ancillary` register 2^{t-1} times. This projects the phase of W onto the `counting` register, where the phase can be mapped directly to the `counting` register by applying the inverse Fourier transform. Performing a measurement on the `counting` register then reveals E_k.

4.2 Implementation in ProjectQ

This algorithm has been implemented for a general Hamiltonian in the open-source Python software framework for quantum computing by Steiger [10]. We've adapted the code for two Hamiltonians, the molecule hydrogen and molecular hydrogen. The full Jordan-Wigner reduced Hamiltonian for the molecule can be found in 2.10, and the Hamiltonian for molecular hydrogen H_2 is

$$H = g_0 \mathbb{1} + g_1 Z_0 + g_2 Z_1 + g_3 Z_0 Z_1 + g_4 Y_0 Y_1 + g_5 X_0 X_1, \tag{4.2}$$

where the coefficients g are functions of the hydrogen-hydrogen bond angle [3]. Both these Hamiltonians are simple enough for a classical computer to easily calulate its eigenstates and eigenvalues by diagonalization. For complex structures, an approximation of the groundstate has to be found via the method explained in 2.2.1 or via Monte Carlo simulation [1]. However, as we have access to the eigenstates classically, we can artificially create an approximation of the ground by creating a superposition of the ground state and an excited state. But before we cover the results of the Hamiltonian simulation, let us illustrate the implementation by taking a trivial Hamiltonian and reducing all the register sizes to one. The following circuit is a drawing of all the operations with the simplified parameters, generated by the python package Matplotlib via ProjectQ.
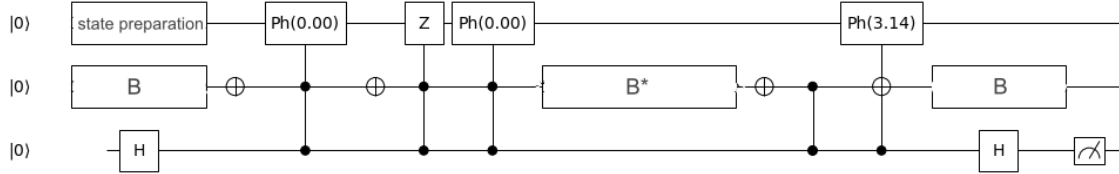
Figure 4.3: Circuit of the Spectrum by quantum walk implemented in ProjectQ, for single-qubit registers and trivial Hamiltonian.

The first register is the `physical` register, the second register is the `ancillary` register and the final register is the `counting` register. While there is some non-conventional, ProjectQ-specific notation in the Pauli operations, this drawing shows that the implementation follows the protocol discussed above.

4.3 Results

Before we show the results of the implementation, let us motivate the state preparation sequence used. As we can obtain the ground state corresponding to the specific Hamiltonian classically, we artificially create an approximation of the ground state by preparing a superposition of the classicaly obtained eigenstates. We first show that the distribution of obtained energies is directly related to the probabilites of the states in the superposition of the approximated ground state. After we have shown this relation, we then try to get energy measurements of H_2 and H as accurately as possible by treating the best case scenario, in which the `physical` register is prepared in the classically obtained ground state. We acknowledge that this is not conform protocol for more complex systems of which we cannot just obtain the eigenstates classically. In section 2.2.1 we show how to prepare an approximation of the ground state in such case. However, the added inaccuracy of non-ground energy measurements we'd introduce is rather trivial, as we can predict the energy distribution anyway. With our precise energy measurements, we aim to show that this algorithm, when the input is the ground state, returns energies that match the exact, well-established values with a certain accuracy.

First we consider the results of energy measurements on an approximation of the groundstate. We prepare the physical register in an equal superposition of the actual ground state and some excited state:

$$\left|\tilde{\phi}\right\rangle = \frac{1}{\sqrt{2}}\left|\phi_0\right\rangle + \frac{1}{\sqrt{2}}\left|\phi_k\right\rangle \tag{4.3}$$

Perfoming a series of iterations of the algorithm result in the energy distribution plotted in **??**. As the distribution of energies is directly related to the probability of the corresponding basis state in the approximation, this result matches our expectation, given the probabilities of $\frac{1}{2}$. The energy corresponding to the second peak is roughly 0.15 Hartree. This matches the first excited
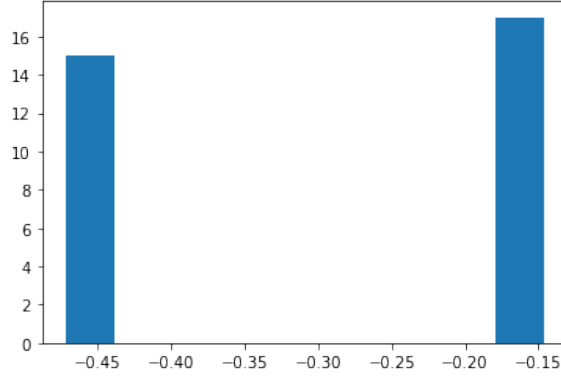
Figure 4.4: Energy distribution of 32 iterations of the algorithm with a phase estimation precision of $t = 8$.

state of the hydrogen atom The ground energy seems close to the theoretical value and we now perform a series of highly precise measurements on the ground state, of which the result is displayed below.
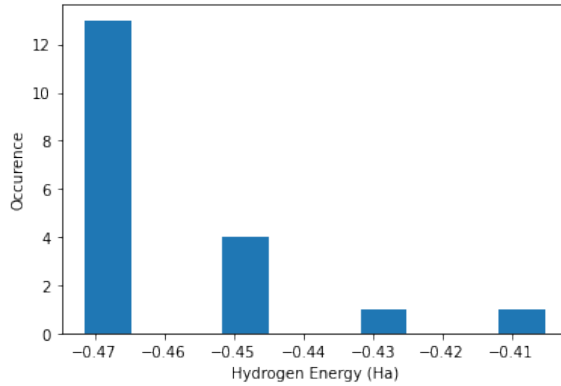

Figure 4.5: A sampling of energy measurements when the physical register is prepared in the ground state.

The small fluctuations seen are due to the present imprecision in the phase-estimation process. The lowest energy sampled is -0.4714 Hartree, with a phase estimation accuracy of $t = 10$. The theoretical value of the ground energy of the hydrogen molecule is -0.5 Hartree. As the Jordan-Wigner mapping is exact and the Born-Oppenheimer approximation is considered in the theoretical value, this inaccuracy is caused by a combination of the limited accuracy of phase estimation and inaccuracies of the state preparation.

Next, we implement the H_2 molecule's Hamiltonian, where each sequence we find the ground energy corresponding to the bond length in Angstrong. We've performed three different sequences of measurements with a respective phase estimation accuracy of $t = 8$, $t = 9$, $t = 10$. The results are plotted below.
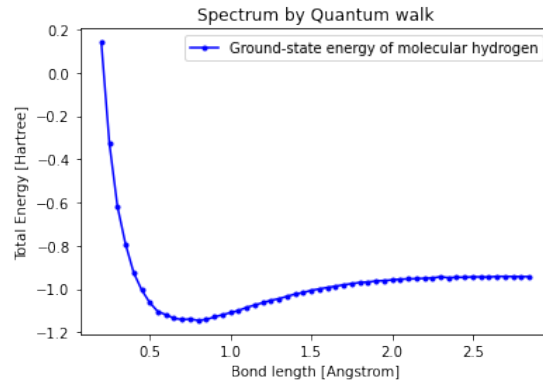
(a) Potential energy of H_2 against bond angle, calculated by the spectrum by quantum walk algorithm for a phase estimation precision of $t = 10$
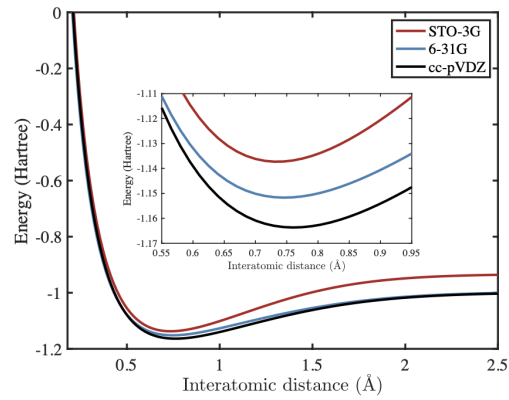
(b) Exact potential curves of H_2 for different basis sets [2, p. 42]

Chapter 5

Discussion and Conclusion

The main drawback of the Spectrum by quantum walk algorithm is that before implementing W, we have to rescale the coefficients of the Hamiltonian by a factor of $\mathcal{N}$. By doing so, we rescale the spectrum as the spectral gap (the difference between the ground state and the first excited state) with that same factor [1, 10]. This is problematic as the resolution of the energy measured is lowered by the rescaling. When compared to Trotterization, which implements an approximation of the time evolution operator $\exp\{-iHt\}$, W needs to be applied $\mathcal{N}t$ times to achieve the level of accuracy obtained by Trotterization [10, p. 87]. Whether implementing W is more efficient than Trotterization depends on the Hamiltonian in question as well as the number of Trotter steps needed to implement the time-evolution operator. Ultimately we aim to simulate large molecules and as we currently do not have the resources to simulate complex Hamiltonians, we are unable to estimate the advantage of the Spectrum by quantum walk algorithm.

put values of G

Bibliography

[1] D. Poulin et al. "Quantum Algorithm for Spectral Measurement with a Lower Gate Count". In: *Physical Review Letters* 121.1 (July 2018). ISSN: 1079-7114. DOI: 10.1103/physrevlett. 121.010501. URL: http://dx.doi.org/10.1103/PhysRevLett.121.010501.

[2] S. McArdle et al. "Quantum computational chemistry". In: *Reviews of Modern Physics* 92.1 (Mar. 2020). ISSN: 1539-0756. DOI: 10.1103/revmodphys.92.015003. URL: http://dx.doi.org/10.1103/RevModPhys.92.015003.

[3] P. J. J. O'Malley et al. "Scalable Quantum Simulation of Molecular Energies". In: *Phys. Rev. X* 6 (3 July 2016), p. 031007. DOI: 10.1103/PhysRevX.6.031007. URL: https://link.aps.org/doi/10.1103/PhysRevX.6.031007.

[4] P. W. Shor. "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer". In: *SIAM Review* 41.2 (1999), pp. 303–332. ISSN: 00361445. URL: http://www.jstor.org/stable/2653075.

[5] M. Reiher et al. "Elucidating reaction mechanisms on quantum computers". In: *Proceedings of the National Academy of Sciences* (2017). ISSN: 0027-8424. DOI: 10.1073/pnas.1619152114. eprint: https://www.pnas.org/content/early/2017/06/30/1619152114.full.pdf. URL: https://www.pnas.org/content/early/2017/06/30/1619152114.

[6] D. W. Berry et al. "Qubitization of Arbitrary Basis Quantum Chemistry Leveraging Sparsity and Low Rank Factorization". In: *Quantum* 3 (Dec. 2019), p. 208. ISSN: 2521-327X. DOI: 10.22331/q-2019-12-02-208. URL: http://dx.doi.org/10.22331/q-2019-12-02-208.

[7] Y. Cao et al. *Quantum Chemistry in the Age of Quantum Computing*. 2018. arXiv: 1812.09976 [quant-ph].

[8] T. Bian et al. "Quantum computing methods for electronic states of the water molecule". eng. In: *Molecular Physics* 117.15-16 (2019), pp. 2069–2082. ISSN: 0026-8976. URL: http://www.tandfonline.com/doi/abs/10.1080/00268976.2019.1580392.

[9] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. 10th. USA: Cambridge University Press, 2011. ISBN: 1107002176.

[10] D. S. Steiger. "Software and Algorithms for Quantum Computing". en. PhD thesis. Zurich: ETH Zurich, 2018. DOI: 10.3929/ethz-b-000322770.

[11] A. Daskin and S. Kais. "A generalized circuit for the Hamiltonian dynamics through the truncated series". In: *Quantum Information Processing* 17.12 (Oct. 2018). ISSN: 1573-1332. DOI: 10.1007/s11128-018-2099-z. URL: http://dx.doi.org/10.1007/s11128-018-2099-z.

[12] J. T. Seeley, M. J. Richard, and P. J. Love. "The Bravyi-Kitaev transformation for quantum computation of electronic structure". In: *The Journal of Chemical Physics* 137.22 (Dec. 2012), p. 224109. ISSN: 1089-7690. DOI: 10.1063/1.4768229. URL: http://dx.doi.org/10.1063/1.4768229.

[13] K. Snizhko, P. Kumar, and A. Romito. *The Quantum Zeno effect appears in stages*. 2020. arXiv: 2003.10476 [quant-ph].

[14]   R. Babbush et al. "Encoding Electronic Spectra in Quantum Circuits with Linear T Complexity". In: *Physical Review X* 8.4 (Oct. 2018). ISSN: 2160-3308. DOI: 10.1103/physrevx.8.041015. URL: http://dx.doi.org/10.1103/PhysRevX.8.041015.

[15]   M. Mottonen et al. *Transformation of quantum states using uniformly controlled rotations*. 2004. arXiv: quant-ph/0407010 [quant-ph].