Alma Mater Studiorum · Università di Bologna

Dipartimento di Fisica e Astronomia "Augusto Righi"
Corso di Laurea in Fisica

Simulation of H_2 molecule through Variational Quantum Eigensolver

Relatore:
Prof.ssa Elisa Ercolessi

Correlatore:
Dott. Riccardo Cioli

Presentata da:
Alessandro Pasquale

Anno Accademico 2023/2024

Abstract

In questa tesi spiego e implemento un metodo computazionale quantistico per la risoluzione dell'equazione di Schrödinger della molecola di idrogeno. In particolare, determino lo stato fondamentale (ground state) e l'energia minima ad esso associata, generando la distanza internucleare corrispondente. Nel primo capitolo espongo le basi del modello a circuiti della computazione quantistica, cioè il qubit, i sistemi di qubit e le operazioni quantomeccaniche e logico-circuitali che è possibile eseguire su di essi. Fatto questo, è necessario "tradurre" le funzioni d'onda e l'Hamiltoniana relative alla modellizzazione quantomeccanica della molecola di idrogeno in oggetti propri del framework della computazione quantistica, cioè qubit, porte logiche quantistiche e osservabili misurabili su computer quantistici. Questo viene esposto nel secondo capitolo. Nel terzo capitolo commento il programma che codifica ed esegue quanto presentato nel secondo capitolo, seguendo le regole stabilite nel primo. Il programma è scritto utilizzando la libreria open-source PennyLane, che fornisce un framework specifico per la computazione quantistica.

This thesis aims to solve the Schroedinger equation for molecular Hydrogen via a quantum computational approach. It is to find the ground state and its minimum energy, yielding the ground state's internuclear distance. In the first chapter, I explain the fundamentals of the circuit model of quantum computation, namely qubits, multi-qubit systems, and the quantum-mechanical circuit operations that can be performed on them. Once this is done, it is necessary to "translate" the wave functions and the Hamiltonian related to the quantum mechanical modeling of the hydrogen molecule into objects specific to the quantum computing framework, namely qubits, quantum logic gates, and observables measurable on quantum computers. This is presented in the second chapter. In the third chapter, I comment on the program encoding and executing what is presented in the second chapter, according to the rules established in the first. The program is written using the open-source library PennyLane, which provides a dedicated framework for quantum computing.

Introduction

This thesis aims to explore how quantum algorithms can be used to solve optimization problems that are interesting in physics and chemistry, such as finding a molecule's ground state. In the first chapter, I introduce the basis of quantum computing, starting from the framework's fundamental object, the qubit. The two-level quantum system is formalized and exposed in its main quantum-mechanical characteristics. The circuit model of quantum computation is provided with "quantum gates"; those are the operators acting on the qubit, or system of more than one qubit, to manipulate them. To physically realize quantum computers, the correspondent quantum hardware, it's crucial to have "quantum universal gates," through which it is possible to express any quantum gate and, thus, any possible sequence of these implementing algorithms. As physicists started to apply quantum mechanical models to computational problems and swapped digital bits for qubits, the fields of quantum mechanics and computer science began to converge. Over the years, experimentalists have constructed small-scale quantum computers using trapped ions, superconductors, photons, topological qubits, etc. Any of these qubit's physical realization faces different challenges, depending on the characteristics of the physical objects involved. For example, topological quantum qubits seem to be resilient to decoherence, i.e., the phenomenon that makes the quantum mechanical features of the system fragile due to interaction with the environment.

In the second chapter we will consider molecular hydrogen, a quantum system that cannot be solved analytically, instead requiring computational methods to estimate its ground-state energy. The Variational Quantum Eigensolver (VQE) is employed for this purpose, optimizing a parameterized quantum circuit to minimize the expectation value of the Hamiltonian. The molecular wave functions are encoded using qubits and Jordan-Wigner mapping, starting from the Hartree-Fock state as an initial approximation. The VQE is a hybrid protocol that exploits a quantum computer to prepare a variational state according to some specific ansatz, and a classical computer that performs the optimization process then refines the ansatz to approximate the molecule's ground state.

In the final part of the thesis, I describe the program that encodes and executes the tasks presented in Chapter 2 according to the framework described in Chapter 1. The implementation was done using the PennyLane library, a fundamental resource for quantum computing applied to various fields. The computation results align with the experimental data of the hydrogen molecule.

Contents

1

Basics of Quantum Computation

In this chapter, I introduce the basis of quantum computing, starting from the framework's fundamental object, the qubit. The two-level quantum system is formalized and exposed in its main quantum-mechanical characteristics. The circuit model of quantum computation is provided with "quantum gates"; those are the operators acting on the qubit, or system of more than one qubit, to manipulate them. Such operators ensure probability conservation by preserving the qubit's normalization; this condition is the operator's unitariness requirement. To physically realize quantum computers, the correspondent quantum hardware, it's crucial to have "quantum universal gates," i.e., through which it is possible to express any quantum gate and, thus, any possible sequence of these (algorithms). Quantum universal gates constitute the elementary, fundamental building blocks of circuital modules, built in quantum computers (similarly to classical ones). Finally, I present the measurement of observables, which is fundamentally different from the classical case. The measurement of an observable is the procedure that, when repeated a significant number of times, allows us to determine the state of the prepared qubit system; it is the statistical approach necessary when dealing with quantum systems. A thorough introduction to these topics can be found in [1].

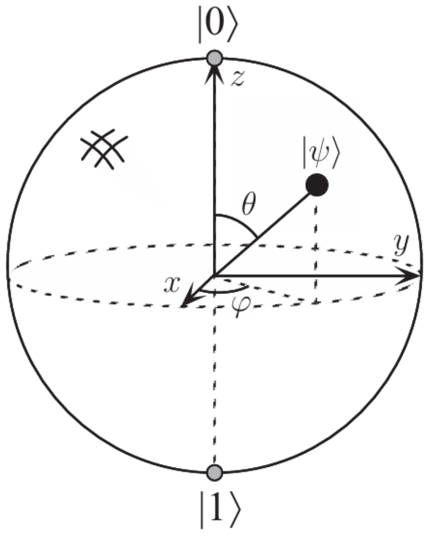
1.1 Quantum bits

The fundamental building block of quantum computation is the two-level quantum system. A two-level quantum system is a system that can access two distinct states. This translates into a wave function that describes the system in a binary superposition. We will denote it:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \tag{1.1}$$

The wave function assumes, in quantum mechanics context, the role of an element in a Hilbert space, that is, a vector space equipped with an inner product. Thus, the qubit case has two basis vectors, matching the dimension of the space where the qubit "resides." The qubit is described by a linear combination of these basis vectors with complex coefficients. Since $|\alpha|^2 + |\beta|^2 = 1$, we may rewrite eq.(1.1) as

$$|\psi\rangle = \cos\frac{\theta}{2}|0\rangle + e^{i\varphi}\sin\frac{\theta}{2}|1\rangle \tag{1.2}$$



h

Figure 1.1. Bloch sphere, image taken from [1]

where $\theta \in [0, \pi[, \quad \varphi \in [0, 2\pi]$. The condition on the coefficients α, β arises from the nature of the wave function, the latter being a probability density function. This leads to the result that every basis of the state ket space corresponds one-to-one with the unit vectors of the unit sphere in $\mathbb{R}^3$, the so-called Bloch sphere fig.(1.1).

1.1.1 Multiple qubits

When there are more objects, each one behaving like a 2-state quantum system, we want the wave function to encode the information of more than one qubit. For example, for two qubits, we have:

$$|\psi\rangle = \alpha_{00}|00\rangle + \alpha_{01}|01\rangle + \alpha_{10}|10\rangle + \alpha_{11}|11\rangle. \tag{1.3}$$

Similar to the case for a single qubit, the measurement result $x = 00, 01, 10,$ or 11 occurs with probability $|\alpha_x|^2$, with the state of the qubits after the measurement being $|x\rangle$. More generally, a state of N qubits is a vector of the tensor Hilbert space $\mathcal{H}_N = \mathcal{H} \otimes \cdots \otimes \mathcal{H} = \mathcal{H}^{\otimes N}$. We're basically operating on the composed Hilbert space of the qubits involved, which requires the tensor product formalism. Given two Hilbert spaces V and W (finite dimensional ones, respectively n and m) we define the vector space of these as $V \otimes W$ which happens to be mn dimensional. The elements of $V \otimes W$ are linear combinations of 'tensor products' $|v\rangle \otimes |w\rangle$ of elements $|v\rangle$ of V and $|w\rangle$ of W. In particular, if $|i\rangle$ and $|j\rangle$ are orthonormal bases for the spaces V and W, then $|i\rangle \otimes |j\rangle$ is a basis for $V \otimes W$. This is still a Hilbert space, with all the features like linearity.

1.2 Quantum circuits

In analogy to the classical case of a bit, logical operations can be performed on a qubit to change its state. As mentioned earlier, the wave function represents a probability distribution. In order to preserve probability, a generic operation should be unitary:

$$U|\psi\rangle = |\psi'\rangle \quad \langle U\psi, U\psi\rangle = \langle \psi, U^\dagger U\psi\rangle = 1 \implies U^\dagger U = \mathbb{I} \tag{1.4}$$

Unitary matrices thus take on the role of quantum logic gates, which take an input state vector and output the result of the operation, that is, a different state. This is the theoretical framework for analyzing and constructing quantum algorithms: a series of operations on one or more qubits with various functions. For example, let's fix the computational basis $|0\rangle, |1\rangle$ and apply the linear unitary operator X.

$$X \equiv \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \tag{1.5}$$

$$X \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \beta \\ \alpha \end{pmatrix} \tag{1.6}$$

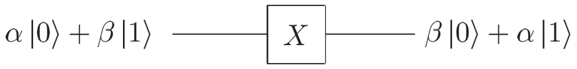


Figure 1.2. A circuit with the X gate, image taken from [1]

1.2.1 Single qubit gates

Let's now take a brief dive that will help us understand operations on the qubit. Since the state ket space is two-dimensional, the operator algebra is four-dimensional [2]. Specifically, every linear operator can be expressed as :

$$\hat{A} = a_m \hat{1} + \mathbf{a} \cdot \hat{\boldsymbol{\sigma}} \tag{1.7}$$

where $\mathbf{a} \cdot \hat{\boldsymbol{\sigma}} = \sum_{i=1}^{3} a_i \hat{\sigma}_i$, $\hat{\sigma i}$ being the pauli operators. $\hat{A}$ is self-adjoint if and only if $a_m \in \mathbb{R}$ and $\mathbf{a} \in \mathbb{R}^3$. Once a basis for $\mathbb{R}^3$ is fixed, a matrix expression for the Pauli operators exists. We will call this basis "auxiliary basis." If we take the Cartesian basis, so that

$\mathbf{a} = (a_x, a_y, a_z)$, the matrix representation of the Pauli operators is:

$$\sigma_x = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad \sigma_y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, \quad \sigma_z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}. \tag{1.8}$$

By exploiting the property of A being self-adjoint, we can take the eigenket basis that results from it. It is fundamental, therefore, to solve the eigenvalue problem for operators of the self-adjoint type $\hat{A}$. Throughout this discussion, we fix the auxiliary basis as the Cartesian basis and take the computational basis $|0\rangle, |1\rangle$ as the basis corresponding to the eigenkets of σ_z. I want to emphasize that every operation can be expressed as the linear combination of the identity and the Pauli matrices; in particular, if the coefficients are real, then the resulting operator is self-adjoint and can be interpreted as an observable. Linear combinations with complex coefficients also exist; even if they do not represent observables, they function as quantum logic gates, provided they are unitary. The Pauli matrices give rise to three useful unitary matrix classes when exponentiated (see appendix (A.1)). These are the rotation operators about the $\hat{x}$, $\hat{y}$, and $\hat{z}$ axes, defined by the equations (we have fixed the cartesian auxiliary base):

$$R_x(\theta) \equiv e^{-i\frac{\theta}{2}\sigma_x} = \cos\left(\frac{\theta}{2}\right) I - i\sin\left(\frac{\theta}{2}\right)\sigma_x = \begin{bmatrix} \cos\left(\frac{\theta}{2}\right) & -i\sin\left(\frac{\theta}{2}\right) \\ -i\sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{bmatrix} \tag{1.9}$$

$$R_y(\theta) \equiv e^{-i\frac{\theta}{2}\sigma_y} = \cos\left(\frac{\theta}{2}\right) I - i\sin\left(\frac{\theta}{2}\right)\sigma_y = \begin{bmatrix} \cos\left(\frac{\theta}{2}\right) & -\sin\left(\frac{\theta}{2}\right) \\ \sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{bmatrix} \tag{1.10}$$

$$R_z(\theta) \equiv e^{-i\frac{\theta}{2}\sigma_z} = \cos\left(\frac{\theta}{2}\right) I - i\sin\left(\frac{\theta}{2}\right)\sigma_z = \begin{bmatrix} e^{-i\frac{\theta}{2}} & 0 \\ 0 & e^{i\frac{\theta}{2}} \end{bmatrix} \tag{1.11}$$

where $\theta \in [0, \pi[$. We can then show that every qubit operation can be written as:

$$U = e^{i\alpha} R_x(\beta) R_y(\gamma) R_z(\delta) \tag{1.12}$$

with $\alpha, \beta, \gamma\, \delta \in \mathbb{R}$ Given $\mathbf{n} = (n_x, n_y, n_z)$ a versor, we can set:

$$R_{\hat{n}}(\theta) \equiv \exp\left(-i\frac{\theta}{2}\boldsymbol{n} \cdot \boldsymbol{\sigma}\right) = \cos\left(\frac{\theta}{2}\right) I - i\sin\left(\frac{\theta}{2}\right)(n_x\sigma_x + n_y\sigma_y + n_z\sigma_z) \tag{1.13}$$

and

$$U = e^{i\alpha} R_{\hat{n}}(\theta) \tag{1.14}$$

That is, each unitary operator corresponds to a direction, an orientation, and an angle. In other words, if such an operator transforms a state vector, it is rotated around $\hat{n}$ by an angle θ. This correspondence also reflects on the possible basis for the ket state space. It is indeed true that given U, a unitary operator acting on that space, this can be diagonalized, finding an orthonormal basis composed of eigenvectors of U (these are determined up to an overall phase factor $e^{i\alpha}$).

1.2.2 Multiple qubit gates

Now, let us generalize from one to multiple qubits. The prototypical multi-qubit quantum logic gate is the controlled-NOT or CNOT gate. This gate (fig. 1.3) has two input qubits, the control and target qubit. The gate's action may be described as follows: if the control qubit is set to 0, then the target qubit is not changed; if the control qubit is set to 1, then the target qubit is flipped. In equations:

$$|00\rangle \rightarrow |00\rangle, \quad |01\rangle \rightarrow |01\rangle, \quad |10\rangle \rightarrow |11\rangle, \quad |11\rangle \rightarrow |10\rangle$$

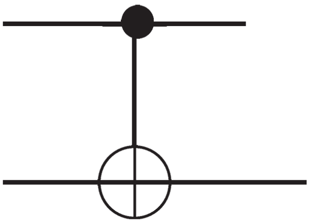


Figure 1.3. Circuital representation of CNOT gate, the control and target qubit are respectively the first and second wire. Image taken from [1]

$$\text{CNOT} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

The controlled-NOT and single qubit gates are the prototypes for all other gates because of the following remarkable universality result: any multiple qubit logic gate can be expressed in terms of CNOT and single-qubit gates. The proof of this result can be found in the next section. If the circuit we're studying is composed of multiple wires, that is, the system has more than one qubit, what sorts of linear operators act on the correspondent tensor space? Suppose $|v\rangle$ and $|w\rangle$ are vectors in V and W, and A and B are linear operators on V and

W, respectively. Then we can define a linear operator $A \otimes B$ on $V \otimes W$ by the equation

$$(A \otimes B)(|v\rangle \otimes |w\rangle) \equiv (A|v\rangle) \otimes (B|w\rangle). \tag{1.15}$$

The tensor product of the Pauli matrices σ_x and σ_y is

$$\sigma_x \otimes \sigma_y = \begin{bmatrix} 0 \cdot \sigma_y & 1 \cdot \sigma_y \\ 1 \cdot \sigma_y & 0 \cdot \sigma_y \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & -i \\ 0 & 0 & i & 0 \\ 0 & -i & 0 & 0 \\ i & 0 & 0 & 0 \end{bmatrix}.$$

Not all operators are of the types $A_i \otimes B_i$, but in general they can be written as $\sum_i A_i \otimes B_i$.

1.2.3 Universal Quantum gates

Two-level unitary gates are universal [1] Consider a unitary matrix U which acts on a d-dimensional Hilbert space. In this section, we explain how U may be decomposed into a product of two-level unitary matrices; that is, unitary matrices that act non-trivially only on two or fewer vector components. The essential idea behind this decomposition may be understood by considering the case when U is 3×3, so suppose that U has the form:

$$U = \begin{bmatrix} a & d & g \\ b & e & h \\ c & f & j \end{bmatrix}. \tag{1.16}$$

We will find two-level unitary matrices $U_1, U_2, \ldots, U_3$ such that

$$U_3 U_2 U_1 U = I. \tag{1.17}$$

It follows that

$$U = U_1^\dagger U_2^\dagger U_3^\dagger. \tag{1.18}$$

U_1, U_2, and U_3 are all two-level unitary matrices, and it is easy to see that their inverses, $U_1^\dagger, U_2^\dagger, U_3^\dagger$, are also two-level unitary matrices. Thus, if we can demonstrate (1.17), then we will have shown how to break U up into a product of two-level unitary matrices. To construct U_1: if $b = 0$, then set

$$U_1 \equiv \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{1.19}$$

If $b \neq 0$, then set

$$U_1 \equiv \begin{bmatrix} \frac{a^*}{\sqrt{|a|^2+|b|^2}} & \frac{b^*}{\sqrt{|a|^2+|b|^2}} & 0 \\ \frac{b}{\sqrt{|a|^2+|b|^2}} & -\frac{a}{\sqrt{|a|^2+|b|^2}} & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{1.20}$$

In either case, U_1 is a two-level unitary matrix. When we multiply the matrices out, we get

$$U_1 U = \begin{bmatrix} a & d & g \\ 0 & e & h \\ c & f & j \end{bmatrix}. \tag{1.21}$$

The key point to note is that the middle entry in the left-hand column is zero. We denote the other entries in the matrix with a generic prime; their actual values do not matter. Now apply a similar procedure to find a two-level matrix U_2 such that $U_2 U_1 U$ has no entry in the bottom-left corner. That is, if $c = 0$, we set

$$U_2 \equiv \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{1.22}$$

If $c \neq 0$, then set

$$U_2 \equiv \begin{bmatrix} \frac{a^*}{\sqrt{|a|^2+|c|^2}} & 0 & \frac{c^*}{\sqrt{|a|^2+|c|^2}} \\ 0 & 1 & 0 \\ \frac{c}{\sqrt{|a|^2+|c|^2}} & 0 & -\frac{a}{\sqrt{|a|^2+|c|^2}} \end{bmatrix}. \tag{1.23}$$

In either case, when we carry out the matrix multiplication, we find that

$$U_2 U_1 U = \begin{bmatrix} 1 & d & g \\ 0 & e & h \\ 0 & f & j \end{bmatrix}. \tag{1.24}$$

Since U, U_1, and U_2 are unitary, it follows that $U_2 U_1 U$ is unitary, and thus $d = g = 0$, since the first row of $U_2 U_1 U$ must have norm 1. Finally, set

$$U_3 \equiv \begin{bmatrix} 1 & 0 & 0 \\ 0 & e^* & f^* \\ 0 & h^* & j^* \end{bmatrix}. \tag{1.25}$$

It is now easy to verify that $U_3U_2U_1U = I$, and thus

$$U = U_1^\dagger U_2^\dagger U_3^\dagger,$$

which is a decomposition of U into two-level unitaries. More generally, suppose U acts on a d-dimensional space. Then, in a similar fashion to the 3×3 case, we can find two-level unitary matrices $U_1, \ldots, U_{d-1}$ such that the matrix

$$U_{d-1}U_{d-2}\cdots U_1U$$

has a 1 in the top-left-hand corner and all zeroes elsewhere in the first row and column. We then repeat this procedure for the $(d-1)\times(d-1)$ unitary submatrix in the lower-right-hand corner of $U_{d-1}U_{d-2}\cdots U_1U$, and so on, with the end result that an arbitrary $d \times d$ unitary matrix may be written

$$U = V_1 \cdots V_k, \tag{1.26}$$

where the matrices V_i are two-level unitary matrices, and $k \leq (d-1) + (d-2) + \cdots + 1 = \frac{d(d-1)}{2}$.

Single qubit and CNOT gates are universal So, any quantum gate can be expressed with at most $\frac{d(d-1)}{2}$ two-level quantum gates, where d is the dimension of the state space the quantum acts on. Using Gray codes [1], we can implement any two-level quantum gate using only single-qubit gates (the coefficients of which are the coefficients of the two-level unitary matrices we use to express the arbitrary unitary matrix) and CNOTs. In other words, any unitary matrix can be implemented using single qubit gates and CNOTs.

A discrete set of universal operations We've established that any quantum gate can be expressed and implemented using single-qubit gates and CNOTs. Now, let's find a discrete set of gates that allows us to approximate all other gates, i.e., any unitary operator, with sufficient accuracy. We'll immediately include the CNOT gate in this set, and we're left to show how any single-qubit gate can be approximated using a discrete set of gates. Consider the following discrete set of gates:

- H is the Hadamard gate,
- T is a $\pi/4$ rotation around the z-axis, defined as:

$$T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}.$$

The general reasoning is structured as follows:

- T is a $\pi/4$ rotation around the z-axis, and HTH is a $\pi/4$ rotation around the x-axis:

  $$R_z(\pi/4) = T, \quad R_x(\pi/4) = HTH. \tag{1.27}$$

- composing these two rotations, we can identify a rotation around a specific axis n by a specific angle θ:

  $$\begin{aligned} \exp\Big(-i\frac{\pi}{8}\sigma_z\Big)\exp\Big(-i\frac{\pi}{8}\sigma_z\Big) &= \Big[\cos\frac{\pi}{8}I - i\sin\frac{\pi}{8}\sigma_z\Big]\Big[\cos\frac{\pi}{8}I - i\sin\frac{\pi}{8}\sigma_x\Big] \\ &= \cos^2\frac{\pi}{8}I - i\Big[\cos\frac{\pi}{8}(\sigma_x+\sigma_z) + \sin\frac{\pi}{8}\sigma_y\Big]\sin\frac{\pi}{8} = R_{\hat{n}}(\theta) \end{aligned} \tag{1.28}$$

  where $\boldsymbol{n} = \left(\cos\frac{\pi}{8}, \sin\frac{\pi}{8}, \cos\frac{\pi}{8}\right)$ through an angle θ defined by $\cos\left(\frac{\theta}{2}\right) \equiv \cos^2\frac{\pi}{8}$.

Such a rotation, when iterated, gives rise to different rotations on the Bloch sphere. The question is: how many of the infinitely many possible rotations are there? The key lies in the angle θ. Indeed, it can be shown to be an irrational multiple of 2π. Consequently, a series of rotations around n by that angle "never closes," managing to approximate any rotation on the Bloch sphere arbitrarily, that is, any single-qubit gate. In other words, the rotations obtained by iterating T and HTH indefinitely are arbitrarily close to any single-qubit unitary operator.

1.3 Measurements

The qubit's wave function is designed to highlight its potential measurement results. For instance, in a state represented as in eq.(1.1), the probability of measuring zero is $|\alpha|^2$, assuming that the measurement basis "aligns" with the one used for the preparation of the qubit. Quantum theory postulates that any observable is in one-to-one correspondence with self-adjoint linear operators.

1.3.1 Measurements in bases other than the computational basis

Imagine a measurement device being "rotated" relative to the computational basis. This means that the states detectable by this device differ from the standard computational states. The following expressions provide a means of expressing an arbitrary single-qubit basis.

$$|0\rangle_{\theta,\varphi} = |0\rangle \cos\left(\frac{\theta}{2}\right) + |1\rangle \sin\left(\frac{\theta}{2}\right) e^{i\varphi} \tag{1.29}$$

$$|1\rangle_{\theta,\varphi} = -|0\rangle \sin\left(\frac{\theta}{2}\right) e^{-i\varphi} + |1\rangle \cos\left(\frac{\theta}{2}\right) \tag{1.30}$$

By re-expressing the qubit's superposition in this new basis, we can determine the corresponding probability amplitudes:

$$\alpha|0\rangle + \beta|1\rangle = |0\rangle_{\theta,\varphi}(\alpha \cos\frac{\theta}{2} + \beta \sin\frac{\theta}{2} e^{-i\varphi}) + |1\rangle_{\theta,\varphi}(-\alpha \sin\frac{\theta}{2} e^{i\varphi} + \beta \cos\frac{\theta}{2})$$

This result extends to systems of multiple qubits, such as those described by the Bell basis [1].

1.3.2 POV measurement

Projective measurements are an essential class of operators encoding measurement in quantum systems. It is a general tool that allows one to describe any observable (its distribution of outcomes) on qubit systems. This fact stems from the property of selfadjoint operators, which possess an orthonormal basis composed of their eigenvectors. Here's the definition [1]:
A projective measurement is described by an observable, M, a self-adhoint operator on the state space of the observed system. The observable has a spectral decomposition:

$$M = \sum_m \lambda_m P_m \tag{1.31}$$

where P_m are called "projectors"; they project onto the respective eigenvector subspace. Given a self-adjoint operator (on a finite dimensional Hilbert space) it is always possible to create the projectors determining the POV measurement. λ_m is the eigenvalue correspondent to the eigenvector; if a measurement procedure yields a state belonging to that eigenspace, λ_m is the numerical value measured. The probability of getting result $\lambda_m$is

$$p(m) = \langle\psi|P_m|\psi\rangle \tag{1.32}$$

Given that outcome λ_m occurred, the state of the quantum system immediately after the measurement is

$$\frac{P_m|\psi\rangle}{\sqrt{p(m)}} \tag{1.33}$$

The average value of M turns out to be

$$E(M) = \sum_m \lambda_m p(m) = \sum_m \lambda_m \langle\psi|P_m|\psi\rangle = \langle\psi|M|\psi\rangle. \tag{1.34}$$

2

VQE for molecular Hydrogen

Molecular Hydrogen [3] is a quantum system that is not analytically solvable. Therefore, we need to use computational methods to obtain the information we want to compare with the experimental data. Specifically, we are looking for an estimate of the value of the ground-state energy of the H_2 molecule, assuming that the nuclei are at rest. A variational method, the Variational Quantum Eigensolver (VQE) [4] is chosen to achieve the goal. In the VQE algorithm, a parametrized circuit is optimized to minimize the expectation value of a Hamiltonian. On paper, we could perform a measurement in the eigenbasis of H, but this requires knowledge of the transformation that diagonalizes it, which is at least as hard as computing the ground-state energy [5]. We first need to encode the molecule wave functions into the quantum computing framework, using qubits and Jordan-Wigner mapping [6, 7, 8]. Then, starting from a reasonable hypothesis, the Hartree-Fock state, the circuit yields a parameterized superposition of electrons states [9]. We will then evaluate a cost function: the observable, namely the Hamiltonian, we want to minimize. The optimization process of this circuit generates the ansatz of the molecule.

2.1 The H2 molecule

Our goal is to solve the time-independent, non-relativistic Schrödinger equation for molecular hydrogen, which is given by:

$$\hat{H}\Psi = E\Psi \tag{2.1}$$

where $\hat{H}$ is the Hamiltonian operator, Ψ is the wavefunction, and E is the total energy of the system. The Hamiltonian consists of the kinetic and potential energy terms for all electrons and nuclei:

$$\hat{H} = -\frac{\hbar^2}{2m_p}\left(\nabla_A^2 + \nabla_B^2\right) - \frac{\hbar^2}{2m_e}\left(\nabla_1^2 + \nabla_2^2\right) + \frac{e^2}{R} - \frac{e^2}{r_{A1}} - \frac{e^2}{r_{B1}} - \frac{e^2}{r_{A2}} - \frac{e^2}{r_{B2}} + \frac{e^2}{r_{12}}$$

where:

- The first and second terms represent the kinetic energy of the two nuclei (protons), with masses m_p;

- The third and fourth terms represent the kinetic energy of the two electrons, with mass m_e;
- The fifth term accounts for the repulsion between the two nuclei, separated by a distance R;
- The sixth-to-ninth terms represent the Coulomb attraction between each electron and each nucleus, where r_{Ai}, r_{Bi} is respectively the distance between electron i and nucleus A and B;
- The tenth term accounts for the repulsion between the two electrons, separated by a distance r_{12}.

This Hamiltonian fully describes the quantum mechanical interactions in the H_2 molecule. Since solving this equation exactly is impossible due to electron correlation and the complexity of the interactions, various approximations and computational methods are employed, which we will discuss in the next sections:

- The **Born-Oppenheimer Approximation** (BOA), which separates nuclear and electronic motion. The mathematical encoding of this approximation is the factorization of the wave function. (Independence ⇔ factorization).
- The **Nuclei** are **at rest**, this enables us to neglect the kinetic energy contribution of the nuclei on the total energy of the system
- **Hartree-Fock** (HF) **Theory**, which approximates the wave function as a single Slater determinant.

2.1.1 Born-Hoppeneimer & Electronic Hamiltonian

To simplify the problem, we apply the BOA in which we treat the nuclei as stationary classical particles. This is justified as the ratio of electronic to nuclear mass is roughly 1:1000, leading to a separation in the time scale of their dynamics. Thus, the molecule wave function can be factorized (the coordinates are referred to as the center-of-mass frame of reference):

$$\psi(\mathbf{r}, \mathbf{R}) = \psi_n(\mathbf{R})\psi_e^{(\mathbf{R})}(\mathbf{r}) \tag{2.2}$$

Here $\mathbf{r}, \mathbf{R}$ encode respectively the electrons and nucleis coordinates shown in fig.2.1 and $\mathbf{R}$ is considered a parameter within ψ_e. The functions ψ_e and ψ_n separately satisfy the eigenvalue equations for H_e and H_n, respectively:

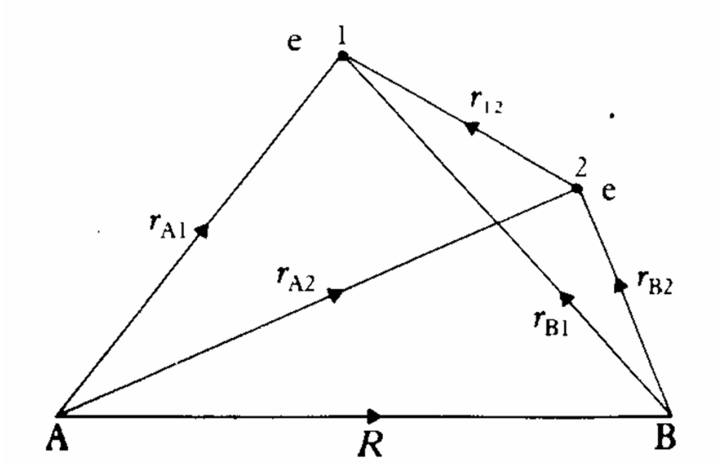


Figure 2.1. Coordinates, image taken from [3]

$$H_n\psi_n(\mathbf{R}) = \left(-\frac{1}{2}\left(\nabla_A^2 + \nabla_B^2\right) + \frac{1}{R}\right)\psi_n(\mathbf{R}) = E_n\psi_n(\mathbf{R}) \tag{2.3}$$

$$\begin{aligned} H_e(R)\psi_e^{(R)}(\mathbf{r}) &= \left\{-\frac{1}{2}\left(\nabla_1^2 + \nabla_2^2\right) - \frac{1}{r_{A1}} - \frac{1}{r_{B1}} - \frac{1}{r_{A2}} - \frac{1}{r_{B2}} + \frac{1}{r_{12}}\right\}\psi_e^{(R)}(\mathbf{r}) \\ &= [E - E_n(R)]\psi_e^{(R)}(\mathbf{r}) = E_e(R)\psi_e^{(R)}(\mathbf{r}) \end{aligned} \tag{2.4}$$

The nuclear energy contribution is thus encoded in $E_e(R)$ and since the nuclei are assumed to be at rest, there is no contribution from the kinetic part of eq.(2.3). Within the Hartree-Fock approximation, the system is described using the independent particle model. In other words, we treat the molecule as being "split" into the two molecular ions H_2^+ that compose it. Consequently, we assume that the ground state of the molecule is a superposition of the ground states of these molecular hydrogen ions. In the ground state, the single electron in a molecular hydrogen ion is described by two orbital wavefunctions called "Gerade" and "Ungerade" [3]:

$$\begin{aligned} \Phi_g^i(r_{Ai}, r_{Bi}) &= \frac{1}{\sqrt{2}}[\phi_{1s}(r_{Ai}) + \phi_{1s}(r_{Bi})] \\ \Phi_u^i(r_{Ai}, r_{Bi}) &= \frac{1}{\sqrt{2}}[\phi_{1s}(r_{Ai}) - \phi_{1s}(r_{Bi})] \end{aligned} \qquad i = 1, 2 \tag{2.5}$$

where $\phi_{1s}(r) = \phi_{n=1,l=0,m_l=0}(r) = \pi^{-1/2}\exp\{(-r)\}$ is the normalized ground state wave function for atomic hydrogen. (n, l, m_l) are respectively the principal, orbital and magnetic quantum number; s is the symbol for the atomic orbital with $l = 0$. χ_{S,M_S} are the spin operators eigenvectors of a two-spin system.

$$\begin{aligned}\chi_{0,0}(1,2) &= \frac{1}{\sqrt{2}}\left[\alpha(1)\beta(2)-\beta(1)\alpha(2)\right]\\ \chi_{1,1}(1,2) &= \alpha(1)\alpha(2)\\ \chi_{1,0}(1,2) &= \frac{1}{\sqrt{2}}\left[\alpha(1)\beta(2)+\beta(1)\alpha(2)\right]\\ \chi_{1,-1}(1,2) &= \beta(1)\beta(2)\end{aligned} \tag{2.6}$$

where $\alpha(1)\beta(2) = |\uparrow\rangle|\downarrow\rangle$. The independent particle approximation allows us to factorize the initial wave function as a product of individual electron wave functions. Molecular hydrogen differs from the ion by the presence of an additional electron. Therefore, we treat the system as a simple superposition of the two electrons, each interacting with both nuclei A and B. The lowest states of H_2 are four slater determinants (the others vanish):

$$\Phi_A(1,2) = \Phi_g(1)\Phi_g(2)\chi_{0,0}(1,2) \tag{2.7a}$$

$$\Phi_B(1,2) = \Phi_u(1)\Phi_u(2)\chi_{0,0}(1,2) \tag{2.7b}$$

$$\Phi_C(1,2) = \frac{1}{\sqrt{2}}[\Phi_g(1)\Phi_u(2)+\Phi_g(2)\Phi_u(1)]\chi_{0,0}(1,2) \tag{2.7c}$$

$$\Phi_D(1,2) = \frac{1}{\sqrt{2}}[\Phi_g(1)\Phi_u(2)-\Phi_g(2)\Phi_u(1)]\chi_{1,M_s}(1,2), \quad M_s = 0,\pm 1 \tag{2.7d}$$

where χ_{S,M_S} are the (**??**). This follows from the Pauli exclusion principle, according to which the 2 electron wave function is a *Slater determinant* $\Psi(q_1,q_2)$, or in other words, an antisymmetric product of individual electron spin orbitals (wave functions with both orbital and spin part). It is important to note that the basis for the ground state of the molecule, or the accessible spin-orbitals for each of the two electrons, is chosen to be:

$$\{\Phi_{g\alpha},\Phi_{g\beta},\Phi_{u\alpha},\Phi_{u\beta}\} \tag{2.8}$$

.

2.2 Second Quantization

How can the wave functions of quantum systems be represented on a quantum computer? The basic building blocks of quantum computation are controllable two-level systems, which we called qubits. How the expectation value of an observable (e.g. the Hamiltonian) can be computed on a quantum computer? We need to translate it into quantum computing observable. Starting from the first question, we must translate the wave function basis given in (2.8). We associate every spin-orbital in (2.8) to a qubit; consequently, we have a 4-qubit basis. Each of these qubits is $|0\rangle$ if the correspondent spin-orbital is empty, $|1\rangle$ if it's occupied. What if there are more than one electron, for example two, in a certain spin-orbital? This is impossible because of the fermionic nature of the electron; thus the maximum number of electrons in a system with 4 possible states is 4, one for each state (spin-orbital), allowing us to represent the system with 4 qubits, so that we have 16 possible combinations:

$$\begin{aligned}
&\text{0 electrons} &:\quad& |0000\rangle \\
&\text{1 electrons} &:\quad& |0001\rangle, |0010\rangle, |0100\rangle, |1000\rangle \\
&\text{2 electrons} &:\quad& |0011\rangle, |0101\rangle, |0110\rangle, |1001\rangle, |1010\rangle, |1100\rangle \\
&\text{3 electrons} &:\quad& |0111\rangle, |1011\rangle, |1101\rangle, |1110\rangle \\
&\text{4 electrons} &:\quad& |1111\rangle
\end{aligned}$$

Since the Hydrogen molecule conservs the number of electrons (two) in the system, we can consider the kets with only two "1" (equivalently two "0"); in this way, the subspace spanned by such a basis is a 6th-dimensional space. This representation thus enables us to link each computational basis state $|i_1 i_2 i_3 i_4\rangle$ to a Slater determinant in eq.(2.7). Each $i_j = 1$ if the spin-orbital j is occupied and 0 otherwise. In this case $j = \{g\alpha, g\beta, u\alpha, u\beta\}$:

$$\Phi_A(1,2) = \Phi_g(1)\Phi_g(2)\chi_{0,0}(1,2) \quad \Leftrightarrow \quad |1100\rangle \tag{2.9a}$$

$$\Phi_B(1,2) = \Phi_u(1)\Phi_u(2)\chi_{0,0}(1,2) \quad \Leftrightarrow \quad |0011\rangle \tag{2.9b}$$

$$\Phi_C(1,2) = \frac{1}{\sqrt{2}}[\Phi_g(1)\Phi_u(2) + \Phi_g(2)\Phi_u(1)]\chi_{0,0}(1,2) \quad \Leftrightarrow \quad |1001\rangle \tag{2.9c}$$

$$\Phi_D^0(1,2) = \frac{1}{\sqrt{2}}[\Phi_g(1)\Phi_u(2) - \Phi_g(2)\Phi_u(1)]\chi_{1,0}(1,2) \quad \Leftrightarrow \quad |0110\rangle \tag{2.9d}$$

$$\Phi_D^1(1,2) = \frac{1}{\sqrt{2}}[\Phi_g(1)\Phi_u(2) - \Phi_g(2)\Phi_u(1)]\chi_{1,1}(1,2) \quad \Leftrightarrow \quad |1010\rangle \tag{2.9e}$$

$$\Phi_D^{-1}(1,2) = \frac{1}{\sqrt{2}}[\Phi_g(1)\Phi_u(2) - \Phi_g(2)\Phi_u(1)]\chi_{1,-1}(1,2) \quad \Leftrightarrow \quad |0101\rangle \tag{2.9f}$$

2.2.1 Annihilation and creation operators for fermions

Consider n sites (numbered 1 through n), each of which can be either empty or occupied by a spinless fermionic particle. Such sites will be called spin-orbitals. The Hilbert space $\mathcal{H}$ of this system, known as Fock space, is spanned by 2^n basis vectors $|i_1, \ldots, i_n\rangle$, where $i_j = 0, 1$ is the occupation number of the jth site. Everything related to fermions can be expressed in terms of fermions annihilation and creation operators $a_j, a_j^\dagger$ $(j = 1, \ldots, n)$. The operator a_j acts on basis vectors as follows [6]:

$$a_j|i_1, \ldots, i_{j-1}, 1, \ldots, i_n\rangle = (-1)^{\sum_{s=1}^{j-1} i_s}|i_1, \ldots, i_{j-1}, 0, \ldots, i_n\rangle. \tag{2.10a}$$

$$a_j|i_1, \ldots, i_{j-1}, 0, \ldots, i_n\rangle = 0. \tag{2.10b}$$

$$a_j^\dagger|i_1, \ldots, i_{j-1}, 0, \ldots, i_n\rangle = (-1)^{\sum_{s=1}^{j-1} i_s}|i_1, \ldots, i_{j-1}, 1, \ldots, i_n\rangle. \tag{2.10c}$$

$$a_j^\dagger|i_1, \ldots, i_{j-1}, 1, \ldots, i_n\rangle = 0. \tag{2.10d}$$

$(-1)^{\sum_{s=1}^{j-1} i_s}$ represents the parity of the j-th orbital: it is positive when there is an even number of occupied spin-orbitals between ϕ_1 and ϕ_{j-1}, and negative if they are odd in number. The annihilation and creation operators have the following anticommutation rules[6]:

$$a_j a_k + a_k a_j = 0 \quad a_j^\dagger a_k^\dagger + a_k^\dagger a_j^\dagger = 0 \quad a_j a_k^\dagger + a_k^\dagger a_j = \delta_{jk}$$

so that, for $k > j$:

$$\begin{aligned}
&a_j^\dagger a_k|\ldots, 0, i_{j+1}, \ldots, i_{k-1}, 0, \ldots\rangle = 0,\\
&a_j^\dagger a_k|\ldots, 0, i_{j+1}, \ldots, i_{k-1}, 1, \ldots\rangle = (-1)^{\sum_{s=j+1}^{k-1} i_s}|\ldots, 1, i_{j+1}, \ldots, i_{k-1}, 0, \ldots\rangle,\\
&a_j^\dagger a_k|\ldots, 1, i_{j+1}, \ldots, i_{k-1}, 0, \ldots\rangle = 0,\\
&a_j^\dagger a_k|\ldots, 1, i_{j+1}, \ldots, i_{k-1}, 1, \ldots\rangle = 0.
\end{aligned}$$

2.2.2 Second-Quantized Hamiltonian

When simulating a molecule, it is easier to use the second-quantization Hamiltonian rather than the first-quantization Hamiltonian (2.4).

$$H(R) = \sum_{pq} h_{pq}(R) a_p^\dagger a_q + \sum_{pqrs} h_{pqrs}(R) a_p^\dagger a_q a_r^\dagger a_s. \tag{2.11}$$

The constants $h_{pq}(R)$ and $h_{pqrs}(R)$ are evaluated using an initial Hartree-Fock calculation and relate the second-quantized Hamiltonian to the first quantized Hamiltonian. They are calculated as follows:

$$h_{pq} \equiv \int dq\, \Phi_p^*(q) \left(-\frac{1}{2}\nabla_r^2 - \frac{1}{|\mathbf{R}_A - \mathbf{r}|} - \frac{1}{|\mathbf{R}_B - \mathbf{r}|} \right) \Phi_q(q) \tag{2.12}$$

$$h_{pqrs} = \int dq_1\, dq_2\, \Phi_p(q_1)\, \Phi_q(q_2)\, \frac{\Phi_r(q_1)\Phi_s(q_2)}{r_{12}}. \tag{2.13}$$

q_i denoting both the spatial and spin coordinates and $p, q, r, s = g\alpha, g\beta, u\alpha, u\beta$ as in (2.5).

2.3 Mapping the problem onto the quantum computer

In this section, we focus on the process of mapping the hydrogen molecule problem onto a quantum computer. The goal is to translate the quantum description of the hydrogen molecule into a quantum computing framework, where the system's states and interactions are represented using qubits and quantum operations. By applying quantum mechanical principles, we aim to simulate the molecule's behavior accurately. We will cover the steps required to model the hydrogen molecule's electronic structure, utilizing quantum algorithms to perform calculations that are essential for understanding chemical bonding and molecular properties. We lay the foundation for utilizing quantum computing to study molecular systems that are intractable with traditional methods.

2.3.1 Representing the molecular Hamiltonian in terms of quantum bits: Jordan Wigner transformation

To map the Hamiltonian (2.11) into the quantum computing object we want to measure is to express it through Pauli operators. We can use the (2.10). The Jordan-Wigner transform is used to express fermionic operators in terms of the Pauli operators $\{\sigma_x, \sigma_y, \sigma_z, 1\}$. The Jordan-Wigner transform is given by [10]:

$$\begin{aligned} a_j &\Leftrightarrow {\sigma_z}^{\otimes(j-1)} \otimes \sigma^+ \otimes 1^{\otimes(N-j)} \\ &= \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}^{\otimes j} \otimes \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}^{\otimes(N-j)} \end{aligned} \tag{2.14a}$$

$$\begin{aligned} a_j^\dagger &\Leftrightarrow {\sigma_z}^{\otimes(j-1)} \otimes \sigma^- \otimes 1^{\otimes(N-j)} \\ &= \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}^{\otimes(j-1)} \otimes \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}^{\otimes(N-j)} \end{aligned} \tag{2.14b}$$

where

$$\sigma^+ \equiv \frac{\sigma_x + i\sigma_y}{2} = |0\rangle\langle 1|, \quad \sigma^- \equiv \frac{\sigma_x - i\sigma_y}{2} = |1\rangle\langle 0|.$$

The qubit state $|0 \ldots 0\rangle$ corresponds to the vacuum state, and the string of σ_z operators preserves the commutation relations in (2.10) since σ_z and $\sigma^\pm$ anti-commute. The qubit operators associated with the creation and annihilation operators acting on the H_2 molecule, according to the Jordan-Wigner transform, are given by:

$$a_1 \Leftrightarrow \sigma^+ \otimes 1 \otimes 1 \otimes 1 \tag{2.15a}$$
$$a_2 \Leftrightarrow \sigma_z \otimes \sigma^+ \otimes 1 \otimes 1 \tag{2.15b}$$
$$a_3 \Leftrightarrow \sigma_z \otimes \sigma_z \otimes \sigma^+ \otimes 1 \tag{2.15c}$$
$$a_4 \Leftrightarrow \sigma_z \otimes \sigma_z \otimes \sigma_z \otimes \sigma^+ \tag{2.15d}$$

$$a_1^\dagger \Leftrightarrow \sigma^- \otimes 1 \otimes 1 \otimes 1 \tag{2.16a}$$
$$a_2^\dagger \Leftrightarrow \sigma_z \otimes \sigma^- \otimes 1 \otimes 1 \tag{2.16b}$$
$$a_3^\dagger \Leftrightarrow \sigma_z \otimes \sigma_z \otimes \sigma^- \otimes 1 \tag{2.16c}$$
$$a_4^\dagger \Leftrightarrow \sigma_z \otimes \sigma_z \otimes \sigma_z \otimes \sigma^- \tag{2.16d}$$

The Jordan-Wigner transformation ensures that the fermionic operators satisfy the correct anti-commutation relations by introducing a string of σ_z operators. Since $\sigma_z|0\rangle = |0\rangle,\ \sigma_z|1\rangle = -|1\rangle$, it acts as a counter for the occupation of the preceding spin-orbitals, which corresponds to the parity (2.10) discussed in the previous section.

Substituting (2.15), (2.16) into (2.11) we get the qubits Hamiltonian. It can be computed for an arbitrary value of R. For example:

$$\begin{aligned}
&H(R = 0.74\ \AA) = \\
&-0.10I(0) + 0.17\sigma_z(0) + 0.17\sigma_z(1) + 0.17(\sigma_z(0) \otimes \sigma_z(1)) \\
&-0.22\sigma_z(2) + 0.12(\sigma_z(0) \otimes \sigma_z(2)) + 0.17(\sigma_z(1) \otimes \sigma_z(2)) \\
&+0.05(\sigma_y(0) \otimes \sigma_x(1) \otimes \sigma_x(2) \otimes \sigma_y(3)) \\
&-0.05(\sigma_y(0) \otimes \sigma_y(1) \otimes \sigma_x(2) \otimes \sigma_x(3)) \\
&-0.05(\sigma_x(0) \otimes \sigma_x(1) \otimes \sigma_y(2) \otimes \sigma_y(3)) \\
&+0.05(\sigma_x(0) \otimes \sigma_y(1) \otimes \sigma_y(2) \otimes \sigma_x(3)) \\
&-0.22\sigma_z(3) + 0.17(\sigma_z(0) \otimes \sigma_z(3)) + 0.12(\sigma_z(1) \otimes \sigma_z(3)) + 0.17(\sigma_z(2) \otimes \sigma_z(3))
\end{aligned} \tag{2.17}$$

where $\sigma_z(i)$ with $i = 0, 1, 2, 3$ means the σ_z gate acting on the i-th qubit. Now, in order to apply the variational method, we have to choose an initial wave function (now finally turned into a Fock vector space, i.e., expressed in qubit basis) tailored for our goal, which is to find ground state energy. We naturally choose this to be $|1100\rangle$, i.e. (2.9a); that is, the spin-orbitals $\Phi_g\alpha\ \ \Phi_g\beta$ are occupied in a spin-singlet.

2.3.2 Givens rotations

Any transformation must conserve the number of particles because the number of electrons in a molecule is fixed. A class of transformations of this type exists that, while preserving the number of particles in the system, maps a certain state into a superposition of the state itself and a state with higher energy. These transformations are called "excitations."

- the number of "ones" is conserved,
- the transformed state differs in terms of energy; more specifically, it is a superposition containing an excited state with respect to the initial one.

Let's give an example:

$$G(\theta) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta/2) & -\sin(\theta/2) & 0 \\ 0 & \sin(\theta/2) & \cos(\theta/2) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

which is an example of a Givens rotation: a rotation in a two-dimensional subspace of a larger Hilbert space. In this case, we perform a Givens rotation in a two-dimensional subspace of the four-dimensional space of two-qubit states. This gate allows us to create superpositions by exchanging the particles between the two qubits. Such transformations can be interpreted as a single excitation, where we view the exchange from $|10\rangle$ to $|01\rangle$ as exciting the electron from the first to the second spin-orbital. In our 16^{th}-dimensional space of four qubit states, we consider Givens rotations applied on the Hartree-Fock state (2.9a) because the gates of the parameterized circuit act on it, since it is the starting circuit state.

$$G_1(\theta)|1100\rangle = \cos\left(\frac{\theta}{2}\right)|1100\rangle + \sin\left(\frac{\theta}{2}\right)|0110\rangle \tag{2.18}$$

$$G_2(\theta)|1100\rangle = \cos\left(\frac{\theta}{2}\right)|1100\rangle + \sin\left(\frac{\theta}{2}\right)|1001\rangle \tag{2.19}$$

$$G^{(2)}(\theta)|0011\rangle = \cos\left(\frac{\theta}{2}\right)|0011\rangle + \sin\left(\frac{\theta}{2}\right)|1100\rangle \tag{2.20}$$

$$G^{(2)}(\theta)|1100\rangle = \cos\left(\frac{\theta}{2}\right)|1100\rangle - \sin\left(\frac{\theta}{2}\right)|0011\rangle \tag{2.21}$$

(2.18) and (2.19) are single excitations exchanging, respectively, the first and third qubit, and the second and fourth qubit. (2.20), (2.21) represent the same Givens rotation, acting on two different qubit states. We note that $|1001\rangle \rightarrow |0101\rangle$ is not an excitation because the energy of the configurations, before and after the transformation, is the same since the spin state does not influence the energy. Givens rotations parameters are the ones we're going to optimize to find the ground state energy. Indeed, we calculate the Hamiltonian expectation value on the quantum state generated from the givens rotations.

2.3.3 VQE

Let's summarize everything that has been discussed in this section:

- We have mapped the four spin-orbitals of the trial ground state, where two electrons reside, onto a qubit basis (2.9).
- We have mapped the second quantization Hamiltonian (2.11) into a computable observable (1.34), (2.17) on a quantum circuit.
- We have developed a class of quantum gates that generates a parametrized wavefunction.

The VQE is a quantum variational algorithm, that is, an algorithm iteratively run until the output satisfies the criteria the problem requires. The initial wave function, in our case, is the Hartree Fock wave function for the ground state hydrogen molecule (2.9a), i.e. a guess of the ground-state wave function. Through Givens Rotations (particle preserving gates), we generate a family of states, parameterized by an array of angles, on which we will estimate the Hamiltonian average. These rotations are implemented via a quantum circuit. The optimization of the parameters is then executed with a classical algorithm (on a classical computer), and the iteration of the procedure with those new parameters is the essence of the hybrid variational method. It will estimate the H_2 molecule ground state.

3

VQE implementation for molecular Hydrogen

PennyLane is an open-source Python framework for quantum programming, built by researchers, for research. It supports quantum computing, quantum machine learning, and quantum chemistry, enabling researchers to build circuits, train models, and run on simulators or hardware[11]. In this chapter, I describe how I used it to accomplish what is described in the previous section. The function encoding the VQE is described; another function is defined to run the VQE for different values of internuclear distance to find the one that minimizes the ground state energy.

3.1 Pennylane program

The program is available at this link: https://github.com/alepsq/vqe/blob/main/vqe_H2.py. It consists of two main components:

- **Variational Quantum Eigensolver (VQE)**: Implements the methodology described in Section 2.3, utilizing the Pennylane library to construct the Hamiltonian, define the Hartree-Fock state, and optimize the ansatz parameters.
- **Nuclear distance minimization**: Defines a function that determines the nuclear separation that minimizes the ground-state energy by evaluating different distances iteratively.

3.2 Variational Quantum Eigensolver (VQE)

The function `run_VQE` is the core component responsible for computing the ground state of the molecular Hamiltonian. The structure is defined as follows:

```python
def run_VQE(coordinates, output="all"):
```

This function executes the standard VQE protocol, which consists of:

1. Defining the molecular Hamiltonian based on the `coordinates` of the nuclei.

2. Constructing a variational ansatz.
3. Optimizing the parameters to minimize the energy expectation value.

The molecular Hamiltonian is of the kind of (2.17), a quantum-computable observable. It is constructed via Jordan-Wigner mapping (sec. 2.3), starting from the second-quantized Hamiltonian (2.11). The ansatz is the result of the applications of Givens rotations (2.3) to the Hartree-Fock state (2.9a).

3.2.1 Definition of the Hamiltonian

The hydrogen molecule's Hamiltonian is generated using Pennylane's quantum chemistry module:

- The function `hydrogen_hamiltonian` constructs the molecular Hamiltonian using atomic coordinates.
- The Hartree-Fock (HF) reference state is initialized via `hf`, which defines the electronic configuration.
- The possible single and double excitations are identified using `qml.qchem.excitations`.

3.2.2 Quantum Circuit Ansatz

The variational circuit applies excitations to the HF state:

- The ansatz is constructed using `qml.AllSinglesDoubles`, which applies parameterized single and double excitations.
- The cost function, implemented as a quantum node (`qml.qnode`), computes the expectation value of the Hamiltonian.

3.2.3 Optimization Procedure

The parameters are optimized iteratively:

- An Adam optimizer (`qml.AdamOptimizer`) updates the parameters at each iteration.
- The optimization terminates when the relative energy change is below a threshold (10^{-5}).

3.2.4 Function Output

The function can return various results based on the `output` argument:

- **ground_state**: The optimized quantum state.
- **final_energy**: The minimum energy obtained.
- **optimized_parameters**: The final ansatz parameters.
- **all**: A dictionary containing all the above.

For example, we call the function `run_VQE` with a specific set of atomic coordinates for the hydrogen molecule. The function computes the ground state energy and optimized parameters. The function call is:

```
coordinates = np.array([[0.0, 0.0, -0.8], [0.0, 0.0, 0.8]])
result = run_VQE(coordinates)
```

The output of this function call contains multiple elements: The computed ground state in the computational basis is:

$$|\psi\rangle = -0.1418\,|0011\rangle - 0.00002\,|0110\rangle - 0.0029\,|1001\rangle + 0.9899\,|1100\rangle \tag{3.1}$$

The optimized energy value is:

$$E_{\text{min}} = -1.128\,\text{Ha} \tag{3.2}$$

where Ha is the "Hartree", the energy unit in the atomic unit system. The length unit in the atomic unit system is the Bohr, which relates to the angstrom as: $1Bohr \approx 0.529\mathring{A}$.

3.3 Exploration of Energy as a Function of Distance

The function `qa_ray` explores the dependence of ground-state energy on the internuclear distance.

```
def qa_ray(R):
```

This function scans a range of distances and calculates the corresponding energy values to identify the equilibrium bond length.

3.3.1 Process Description

- A list of nuclear separations R is provided.

- For each value of R, the hydrogen molecule's coordinates are updated.
- The function `run_VQE` is called to determine the ground-state energy and optimal parameters for that specific R.
- The results are stored, and the distance corresponding to the minimum energy is identified.

Stopping Criterion

The iteration stops when the relative energy difference between two consecutive calculations is below a threshold:

$$\frac{|E_i - E_{i-1}|}{E_{i-1}} < 10^{-5}. \tag{3.3}$$

3.3.2 Final Results

The function outputs:

- A list of ground-state wavefunctions for each R.
- The corresponding energy values.
- The optimized parameters for each distance.

3.4 Results

Implementing the VQE algorithm allows for an effective approximation of the ground state of the hydrogen molecule. Figure 3.1 summarizes the results on the energy of the ground state as a function of the internuclear distance R. The energy first decreases and then increases toward a horizontal asymptote. The analysis of energy as a function of distance reveals that the minimum energy occurs at:

$$R_{min} = 0.735\,\mathring{A} \tag{3.4}$$

The correspondent energy is:

$$E_{\mathrm{min}} = -1.137\,\mathrm{Ha} \tag{3.5}$$

while the state is:

$$|\psi\rangle = -0.114\,|0011\rangle - 0.003\,|0110\rangle - 0.002\,|1001\rangle + 0.993\,|1100\rangle \tag{3.6}$$

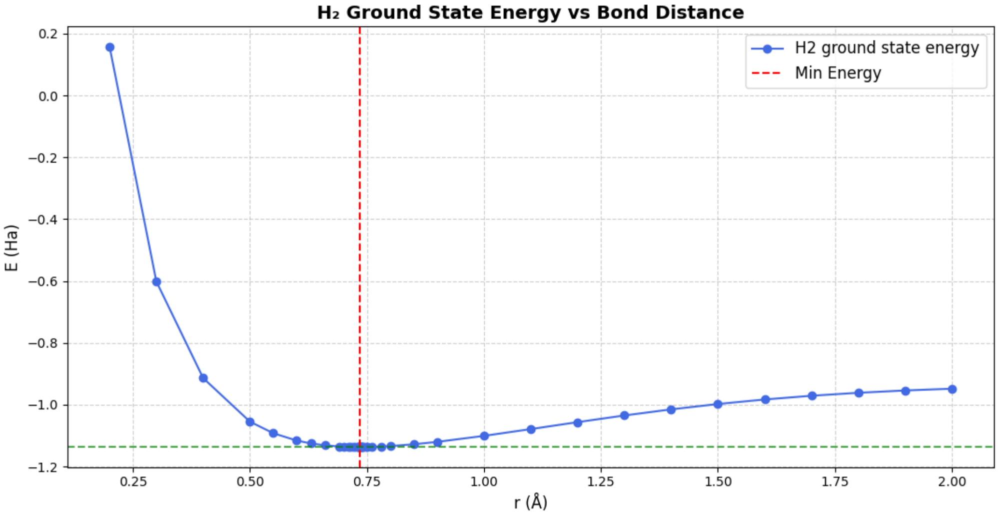


Figure 3.1. Energy as a function of internuclear distance

Conclusion

The objective of this thesis is to demonstrate a straightforward application of quantum computing in the field of condensed matter physics. The results align with both theoretical predictions and experimental measurements, offering an alternative approach to achieving the same outcomes. Computing is a crucial tool in physical research and beyond, with quantum computing providing solutions to specific problems, as exemplified in this work. The Pennylane library serves as an excellent tool for learning how to develop quantum algorithms while simultaneously deepening one's understanding of quantum mechanics, as it offers practical applications of fundamental concepts. The physical realization of quantum hardware remains in development, facing numerous challenges, particularly in maintaining a sufficiently large number of qubits for an adequate period. Despite these ongoing difficulties, many prominent researchers are actively advancing the field, promising significant future progress.

I

Appendix

A

Appendix

A.1 Pauli matrices exponentiated

Let x be a real number and A a matrix such that $A^2 = I$. We want to show that

$$\exp(iAx) = \cos(x)I + i\sin(x)A. \tag{A.1}$$

Proof: The exponential of a matrix A is defined by its power series expansion:

$$\exp(iAx) = \sum_{n=0}^{\infty} \frac{(iAx)^n}{n!}. \tag{A.2}$$

Since $A^2 = I$, we have:

$$A^n = \begin{cases} I & \text{if } n \text{ is even,} \\ A & \text{if } n \text{ is odd.} \end{cases} \tag{A.3}$$

Thus, we can rewrite the power series as follows:

$$\exp(iAx) = \sum_{n=0}^{\infty} \frac{i^n A^n x^n}{n!}. \tag{A.4}$$

Now, separate the terms for even and odd powers of n:

- for even $n = 2k$, $A^{2k} = I$, and the sum becomes:

$$\sum_{k=0}^{\infty} \frac{i^{2k} x^{2k}}{(2k)!} = \sum_{k=0}^{\infty} \frac{(-1)^k x^{2k}}{(2k)!} = \cos(x). \tag{A.5}$$

- for odd $n = 2k+1$, $A^{2k+1} = A$, and the sum becomes:

$$\sum_{k=0}^{\infty} \frac{i^{2k+1} x^{2k+1}}{(2k+1)!} = iA \sum_{k=0}^{\infty} \frac{(-1)^k x^{2k+1}}{(2k+1)!} = iA\sin(x) \tag{A.6}$$

Therefore, combining these results:

$$\exp(iAx) = \cos(x)I + i\sin(x)A. \tag{A.7}$$

Since X, Y, Z are such that $X^2 = 1$ this result holds for them as well.

Bibliography

[1] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2010.

[2] Roberto Zucchini. *Quantum Mechanics: Lecture Notes*.

[3] B. H. Bransden and C. J. Joachain. *Physics of Atoms and Molecules*. Pearson Education, 2003.

[4] Dmitry A. Fedorov, Bo Peng, Niranjan Govind, and Yuri A. Alexeev. Vqe method: A short survey and recent developments. *arXiv:2103.08505*, 2021. URL https://arxiv.org/abs/2103.08505.

[5] Juan Miguel Arrazola. Quantum computing for quantum chemistry: A brief perspective, 2021. URL https://pennylane.ai/blog/2021/11/quantum-computing-for-quantum-chemistry-a-brief-perspective/.

[6] Sergey Bravyi and Alexei Kitaev. Fermionic quantum computation. *arXiv:quant-ph/0003137*, 2000. URL https://arxiv.org/abs/quant-ph/0003137.

[7] Jacob T. Seeley, Martin J. Richard, and Peter J. Love. The bravyi-kitaev transformation for quantum computation of electronic structure. *The Journal of Chemical Physics*, 2012. doi: 10.1063/1.4768229. URL https://pubs.aip.org/aip/jcp/article/137/22/224109/194931/The-Bravyi-Kitaev-transformation-for-quantum.

[8] Yudong Cao, Jonathan Romero, Jonathan P. Olson, Matthias Degroote, Peter D. Johnson, Mária Kieferová, Ian D. Kivlichan, Tim Menke, Borja Peropadre, and Nicolas P. D. Sawaya. Quantum chemistry in the age of quantum computing. *Chemical Reviews*, 119(19), 2019. URL https://pubs.acs.org/doi/10.1021/acs.chemrev.8b00803.

[9] Juan Miguel Arrazola. Givens rotations, 2024. URL https://pennylane.ai/qml/demos/tutorial_givens_rotations.

[10] James D. Whitfield, Jacob Biamonte, and Alán Aspuru-Guzik. Simulation of electronic structure hamiltonians using quantum computers. 2010. URL https://arxiv.org/abs/1001.3855v3.

[11] Pennylane: A quantum machine learning library. URL https://pennylane.ai/.