



UNIVERSIDAD DE INVESTIGACIÓN DE TECNOLOGÍA EXPERIMENTAL YACHAY

Escuela de Ciencias Físicas y Nanotecnología

**TÍTULO: Development and Implementation of a
Universal Quantum Simulator using the Qiskit
platform.**

Trabajo de integración curricular presentado como
requisito para la obtención del título de Físico

Autor:

Cofre Caiza Gabriel Alexander

Tutor:

Ph.D. Guevara Granizo Marco

Co-tutor:

Ph.D. Ph.D. Svozilik Jiri

Urucuí, Junio 2022

SECRETARÍA GENERAL
(Vicerrectorado Académico/Cancillería)
ESCUELA DE CIENCIAS FÍSICAS Y NANOTECNOLOGÍA
CARRERA DE FÍSICA
ACTA DE DEFENSA No. UITEY-PHY-2022-00006-AD

A los 21 días del mes de junio de 2022, a las 09:00 horas, de manera virtual mediante videoconferencia, y ante el Tribunal Calificador, integrado por los docentes:

Presidente Tribunal de Defensa	Dr. PINTO ESPARZA, HENRY PAUL , Ph.D.
Miembro No Tutor	Dr. MOWBRAY , DUNCAN JOHN , Ph.D.
Tutor	Dr. GUEVARA GRANIZO, MARCO VINICIO , Ph.D.

El(la) señor(ita) estudiante **COFRE CAIZA, GABRIEL ALEXANDER**, con cédula de identidad No. **1726385618**, de la **ESCUELA DE CIENCIAS FÍSICAS Y NANOTECNOLOGÍA**, de la Carrera de **FÍSICA**, aprobada por el Consejo de Educación Superior (CES), mediante Resolución **RPC-SO-39-No.456-2014**, realiza a través de videoconferencia, la sustentación de su trabajo de titulación denominado: **Development and Implementation of a Universal Quantum Simulator using the Qiskit platform.** , previa a la obtención del título de **FÍSICO/A**.

El citado trabajo de titulación, fue debidamente aprobado por el(los) docente(s):

Tutor	Dr. GUEVARA GRANIZO, MARCO VINICIO , Ph.D.
--------------	--

Y recibió las observaciones de los otros miembros del Tribunal Calificador, las mismas que han sido incorporadas por el(la) estudiante.

Previamente cumplidos los requisitos legales y reglamentarios, el trabajo de titulación fue sustentado por el(la) estudiante y examinado por los miembros del Tribunal Calificador. Escuchada la sustentación del trabajo de titulación a través de videoconferencia, que integró la exposición de el(la) estudiante sobre el contenido de la misma y las preguntas formuladas por los miembros del Tribunal, se califica la sustentación del trabajo de titulación con las siguientes calificaciones:

Tipo	Docente	Calificación
Miembro Tribunal De Defensa	Dr. MOWBRAY , DUNCAN JOHN , Ph.D.	9,1
Tutor	Dr. GUEVARA GRANIZO, MARCO VINICIO , Ph.D.	10,0
Presidente Tribunal De Defensa	Dr. PINTO ESPARZA, HENRY PAUL , Ph.D.	10,0

Lo que da un promedio de: **9.7 (Nueve punto Siete)**, sobre 10 (diez), equivalente a: **APROBADO**

Para constancia de lo actuado, firman los miembros del Tribunal Calificador, el/la estudiante y el/la secretario ad-hoc.

COFRE CAIZA, GABRIEL ALEXANDER
Estudiante



Firmado electrónicamente por:
GABRIEL ALEXANDER COFRE CAIZA

Dr. PINTO ESPARZA, HENRY PAUL , Ph.D.
Presidente Tribunal de Defensa

HENRY PAUL PINTO ESPARZA

Digitally signed by
HENRY PAUL PINTO ESPARZA
Date: 2022.06.29
17:09:17 -05'00'

Dr. GUEVARA GRANIZO, MARCO VINICIO , Ph.D.
Tutor



Firmado electrónicamente por:
MARCO VINICIO GUEVARA GRANIZO

Dr. MOWBRAY , DUNCAN JOHN , Ph.D.
Miembro No Tutor



Firmado electrónicamente por:
DUNCAN JOHN MOWBRAY .



Firmado electrónicamente por:
**TANIA DANIELA
DE LA CRUZ
ZAPATA**

DE LA CRUZ ZAPATA, TANIA DANIELA
Secretario Ad-hoc

AUTORÍA

Yo, **Gabriel Alexander Cofre Caiza**, con cédula de identidad 1726385618, declaro que las ideas, juicios, valoraciones, interpretaciones, consultas bibliográficas, definiciones y conceptualizaciones expuestas en el presente trabajo; así cómo, los procedimientos y herramientas utilizadas en la investigación, son de absoluta responsabilidad de el/la autora (a) del trabajo de integración curricular. Así mismo, me acojo a los reglamentos internos de la Universidad de Investigación de Tecnología Experimental Yachay.

Urcuquí, Junio 2022.

Gabriel Alexander Cofre Caiza
CI: 1726385618

AUTORIZACIÓN DE PUBLICACIÓN

Yo, **Gabriel Alexander Cofre Caiza**, con cédula de identidad 1726385618, cedo a la Universidad de Investigación de Tecnología Experimental Yachay, los derechos de publicación de la presente obra, sin que deba haber un reconocimiento económico por este concepto. Declaro además que el texto del presente trabajo de titulación no podrá ser cedido a ninguna empresa editorial para su publicación u otros fines, sin contar previamente con la autorización escrita de la Universidad.

Asimismo, autorizo a la Universidad que realice la digitalización y publicación de este trabajo de integración curricular en el repositorio virtual, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior

Urcuquí, Junio 2022.

Gabriel Alexander Cofre Caiza
CI: 1726385618

Dedicatoria

Para ti, que en algún momento coincidimos, hablamos, o simplemente escuchaste de mi.
¡Gracias por tomarte un tiempo en revisar este trabajo, y espero lo disfrutes en gran manera!

Gabriel

Agradecimiento

Agradezco el uso de los servicios de *IBM Quantum Services* para el desarrollo de este trabajo

Gracias a las personas que me han apoyado durante estos años, especialmente a mi tutor: Dr. Jiri Svozilik, quien ha sido un excelente tutor de tesis, profesor, y guía en este proceso de investigación. Gracias por todo el conocimiento, tiempo y energía compartida a lo largo de este proceso sin importar cuán adversas las circunstancias se tornaban. Gracias por impulsarme en este camino cuántico, y brindarme una perspectiva de impacto mundial.

Gracias a la comunidad Yachay, profesores, administrativos, personal de apoyo, y estudiantes. Creo que lo que hemos construido como comunidad nos ha convertido en algo único en el país. Sigamos trabajando, soñando, y colaborando por un país mejor de mano de la ciencia y tecnología. A mis amigos, gracias por todo el apoyo en cada actividad, idea, o iniciativa propuesta.

Gracias a mis padres: Marianita y Norberto, hermanos: Ruth y David, y familia extendida. Infinitas gracias por su respaldo en cada decisión que he tomado independientemente del resultado final. Los quiero, respeto y admiro por su amor incondicional.

Finalmente, creo que he sido afortunado por poder coincidir con todas las personas descritas anteriormente, cada cosa que hemos podido hacer no ha sido coincidencia. Creo que todo es gracia de Dios. Por esta razón, a Él sea toda la gloria, majestad, dominio y autoridad por siempre.

Gabriel

There is plenty of room at the bottom

-Richard Feynman

Resumen

Las computadoras cuánticas son de gran interés para entender los sistemas cuánticos debido a sus ventajas para abordar problemas que las computadoras clásicas no podrían debido a sus diferencias computacionales. Así, una de las líneas de investigación más prometedoras dentro de este campo es la simulación cuántica. En este trabajo de tesis desarrollamos e implementamos un Simulador Cuántico Universal (UQS) en la plataforma Qiskit capaz de implementar el circuito cuántico de cualquier sistema físico, el cual se representa como una suma de potenciales de contribución local. Para probar la viabilidad del UQS, hemos implementado dos sistemas físicos bien estudiados: el Oscilador Armónico Cuántico y el Potencial de Doble Pozo; se comparó la implementación de los circuitos finales, la evolución de sus estados y la profundidad de sus circuitos con trabajos de investigación anteriores. Todos estos sistemas fueron ejecutados en el simulador *statevector*, simulador sin ruido, y luego ejecutados por computadoras cuánticas reales: *ibmq_quito* e *ibmq_lima*, proporcionadas por *IBM-Q*. Las posiciones de estado experimentales obtenidas de chips cuánticos reales se compararon con el simulador silencioso para verificar la fidelidad de los estados utilizando su definición clásica. Como resultado obtuvimos que debido a la aproximación de las funciones de Walsh, el proceso de trotterización y el tiempo de coherencia experimental de cada puerta, existe una cantidad considerable de ruido que no permite notar la evolución del sistema como en el simulador sin ruido.

Palabras Clave: Computación cuántica, simulación cuántica, fidelidad, funciones de Walsh, Simulador Cuántico Universal.

Abstract

Quantum Computers are of great interest for understanding quantum systems due to their advantages in tackling problems that classical computers could not because of their computational differences. Thus, one of the most promising research lines within this field is quantum simulation. In this thesis work, we develop and implement a Universal Quantum Simulator (UQS) in the Qiskit platform capable of implementing the quantum circuit of any physical system, which is represented as a sum of local contribution potentials. To test the feasibility of the UQS, we have implemented two well-studied physical systems: Quantum Harmonic Oscillator and the Double Well Potential; and compared their final circuit implementation, state evolution, and circuit depth with previous research work. All these systems were executed by the state vector simulator, noiseless simulator, and then run by real quantum computers: ibmq-quito and ibmq-lima, provided by IBM-Q. The experimental state positions obtained from real quantum chips were compared with the noiseless simulator to verify the states' fidelities using the classical definition. As a result, we obtained that due to the Walsh Functions approximation, Trotterization process, and experimental coherence time of each gate, there exists a considerable amount of noise that does not allow us to notice the system evolution as in the noiseless simulator.

Keywords: quantum computing, quantum simulation, fidelity, Walsh Functions, Universal Quantum Simulator.

Contents

List of Figures	viii
List of Tables	x
1 Introduction	1
1.1 Problem Statement	2
1.2 General and Specific Objectives	2
2 Theoretical Background	5
2.1 Historical Review	5
2.2 Quantum Computation	6
2.2.1 The qubit	6
2.2.2 System of Multiple Qubits	7
2.2.3 Quantum Gates	8
2.2.4 Measurement	10
2.3 Quantum Simulation	10
2.3.1 From Feynman to Lloyd	12
2.3.2 Universality of the Hamiltonian’s Simulation	12
2.3.3 Quantum Fourier Transform	16
2.3.4 Initialization	16
2.4 IBM Quantum Computers	16
3 Methodology	19
3.1 Algorithm Development	19
3.1.1 Time Evolution Algorithm Design	19
4 Results & Discussion	23
4.1 Calibration	23
4.2 Circuit Implementation	25

4.3	Double Well Potential	30
4.3.1	Discussion	31
4.4	Error Analysis	33
5	Conclusions & Outlook	39
A	Quantum Fourier Transform Gate Representation	41
B	Simulation of the Quantum Harmonic Oscillator on Matlab	43
C	Additional Figures	45
	Bibliography	47

List of Figures

2.1	The Energy States of a Particle used as a Qubit	7
2.2	Bloch Sphere	8
2.3	Quantum circuit symbol for measurement	10
2.4	First Four Walsh Functions	13
2.5	Walsh Function Gate Implementation	15
2.6	Walsh Function w_7 Gate Implementation	15
2.7	Quantum Computer Hardware	17
3.1	4 – <i>qubit</i> Quantum Fourier Transform	20
3.2	4 – <i>qubit</i> Inverse Quantum Fourier Transform	20
3.3	Momentum Gate Implementation	21
3.4	Circuit Gate Modular Implementation, at the end, all qubits are measured.	21
4.1	Operators Approximation with $\Delta t = 0.01$ [ns] using the Walsh decomposition.	24
4.2	Potential and Kinetic Operators approximated using the Walsh decomposition.	24
4.3	Potential and Kinetic Operators approximated in Walsh Function Basis	24
4.4	4th-qubit position Circuit Gate Implementation	25
4.5	Quantum Harmonic Oscillator Evolution in the Statevector Simulator with $\Delta t = 0.01$ [ns]	26
4.6	Quantum Harmonic Oscillator Evolution in the Statevector Simulator with $\Delta t = 0.1$ [ns]	27
4.7	Quantum Harmonic Model using Matlab	28
4.8	Harmonic Oscillator Evolution State’s Probabilities	29
4.9	Fidelity along Time Evolution	30
4.10	Harmonic Oscillator Performance Indicators	30
4.11	Double Potential Well Approximation	31
4.12	Double Potential Well Evolution in the Statevector Simulator with $\Delta t = 0.1$ [ns] and $t = 40$ [ns]	32
4.13	Double Well Potential Evolution	35
4.14	Double Well Potential Fidelity along Time Evolution	36
4.15	Double Well Potential Performance Indicators	36
4.16	QHO Modular Circuit Implementation	37

4.17 Double Well Potential Modular Circuit Implementation	37
4.18 Transpiled Circuits	38
C.1 Operator Approximation Error Distributions for 6 and 7 qubits.	46

List of Tables

2.1	Two-qubit gate Controlled NOT.	9
2.2	Technical Characteristics of IBM-Q processors, adapted from the IBM Quantum Experience, adapted from Ref. 1	18
4.1	Operator's Error Approximation	25
4.2	Measured fidelity fluctuations of the Quantum Harmonic Oscillator and the Double Well Potential .	34

Chapter 1

Introduction

In the last few years, it has been advertised that quantum computers will revolutionize our world. This statement has become familiar and quite abstract, at the same time. Mainly due to the ongoing hype of quantum computers and, unfortunately, the lack of knowledge that most people have about this field, respectively. Indeed, in the Ecuadorian academia, the quantum computing has started to be used as a research line a couple of years ago. Whereas, there has been considerable interest in countries like the United States of America, United Kingdom, and others since a couple of decades ago. However, how valid is the predicted quantum computing revolution? And why must Ecuador be interested in this kind of technology?

Before answering these questions, let us define roughly a *quantum computer*. According to Perez-Delgado and Kok in Ref. 2, a quantum computer is a device that can run a quantum algorithm efficiently, where a quantum algorithm is a classical bit string that encodes a series of quantum operations (typically quantum gates). The quantum computer should be able to take this string as an input and produce another bit string as an output. The probability distribution of the result should be consistent with the predictions of quantum theory. Finally, the time it takes the computer to produce the result should agree with the algorithm's difficulty. Let us consider Shor's factoring as in Ref. 3. It is a quantum algorithm used to find prime numbers of an integer. The number of steps in the algorithm scales polynomially with the input size, but the actual classical implementation will rise exponentially. And as the other example we can think the boson sampling, see Ref. 4. These are some examples among others where classical computers enter to difficulties.

Even though that we can not say that quantum computers have already overcome classical computers, there exist certain areas that play a crucial technological role in the near and extended future that is expected to be revolutionized by quantum computing. Above mentioned Shor's algorithm is one of the most promising results of quantum computing applications, because it enables us to crack the Rivest-Shamir-Adleman (RSA) encryption in a reasonable short time, see Ref. 5. This kind of encryption is used for banking, internet, block-chain, and other industries. Additionally, It has led to the development of new encryption techniques based on quantum mechanics as shown in Refs. 6–8, as well as in other fields, like in finances Refs. 9–11 and optimization problems Refs. 12–14.

As a result, It is mandatory that Ecuador being involved in the so-called quantum-computing race from the

different transversal fields: experimental, theoretical, or computational; presented in the Ref. 2. Being involved in fundamental and applied research will allow us to be knowledge generators in an area of high demand, and high funding. Furthermore, it could support the development of our industry. Especially the ones related to retail and agriculture, as their optimization features could benefit from finding new ways to increase their productivity with fewer resources. Thus, quantum computing is a unique opportunity for Ecuador.

1.1 Problem Statement

Although the research scope in quantum computing is entirely extending, we will focus on simulating quantum physical systems. Actually, within this area, one of the challenges, and the problem that is meant to be tackled in this thesis, is the implementation of arbitrary Hamiltonian evolution via quantum circuits.

We believe that it is essential not only because it allows us to understand the physical systems in research better and compare them with their classical analogs but also to simulate physical systems that are hard, or almost impossible, to do in classical computers because of the computational resources.

Our motivation was to develop and implement a universal quantum simulator implementing a physical system by using the Qiskit platform and testing some simple physical systems. The demonstrations will be run by the real quantum computers of IBMQ and their simulation environments like in Ref. *statevector – simulator* 15. In addition, we compare the evolved states with previously reported research works.

1.2 General and Specific Objectives

The general objective of this thesis project is to design and implement a Universal Quantum Simulator (UQS) in the Qiskit platform that could be used on simulating physical quantum systems in IBM quantum computers. Therefore, this thesis has five chapters, and the structure of each one is the following:

- In the second chapter, Theoretical Background, we make a historical overview of quantum computing and related fields; then, we review key concepts of quantum computing: the qubit, quantum gates, foundation of quantum simulation, how to make a universal quantum simulator using Walsh operators, and we finish with some technical characteristics of IBM quantum computers used in this thesis.
- In the third chapter, Methodology. We start by explaining the general algorithm structure used in the simulator and explain the circuit implementation of each module (Kinetic, Potential operator, Quantum Fourier Transform, and its inverse) in the qiskit platform.
- In the fourth chapter, Results, and Discussion, we show the simple physical systems simulated with the UQS: Quantum Harmonic Oscillator (QHO) and the Double Well Potential (DWP), using the state vector simulator the ibmq-quito and ibmq-lima quantum computers. Additionally, we measure the fidelity with their classical definition and set some calibration parameters of the number of qubits to get a reduced error in the simulation.

-
- In the fifth chapter, Conclusions and Outlook, we show the operational performance of the UQS using the QHO and DWP physical systems. Finally, we provide a future outlook for this research work predominantly focused on increasing the fidelity of executing our simulation in real quantum processors.

Chapter 2

Theoretical Background

2.1 Historical Review

Quantum mechanics is one of the most intriguing physical theories ever developed because of the different interpretations that it may have at the same time. For instance, Bohr believed that particles have probability amplitudes if not observed. Nonetheless, it caused much discussion in the scientific society of those times, mainly because: First was wave-particle duality. Light can act like particles, and particles such as electrons interfere like light waves. According to Bohr, a system behaves as a wave or a particle depending on the context. Second, Heisenberg showed that the uncertainty, for instance, about a particle's position and momentum, is hard-wired into physics. Third, Bohr argued that we could have only probabilistic knowledge of a system: in the Schrödinger's thought experiment, a cat in a box is both dead and alive until is seen. Fourth, particles can become entangled. For example, two spin particles might be in a superposition of their compound states. No matter how far apart they are: if you measure one, you instantly know the state of another one, as in Ref. 16 explains.

All of these features gave birth to different interpretations. The Copenhagen interpretation supported by Bohr stated that quantum mechanics is intrinsically indeterministic and the probabilities are only one thing we are able to acquire from an experiment about the system of interest. Indeed, quantum mechanics is a highly effective tool for predicting measurement results that takes the configuration of the measuring apparatus as input and produces probabilities for the possible measurement outcomes.

Additionally, we have the Many-worlds interpretation stated by Hugh Everett, who proposed a radically new way of interpreting the quantum state. He proposed to take quantum mechanics as descriptive and universal; the quantum state is a genuine description of the physical system concerned, and macroscopic systems are just as well described in this way as microscopic ones. This interpretation immediately solves the "cut" between the micro and macro worlds, and the explanation of particle interference in terms of waves is retained. This interpretation led us to a measurement that results in multiple outcomes occurring in various branches of reality. The complexity of these various scenarios led us to think of extra variables describing the actual state of the world, the hidden variable interpretation proposed by Einstein. Nonetheless, the Bell theorem has offered a tool to test that no hidden-variable completion of quantum

mechanics is not necessary, as in Ref. 17.

In 1969, It was demonstrated, in principle, that quantum systems could be isolated as in Ref. 18 with the invention of the first 3-dimensional optical trap for single neutral particles as in Ref. 18. It helped scientists to explore Nature in a new regime. After three decades and a series of experimental and theoretical discoveries, in 1998, Isaac Chuang from Los Alamos National Laboratory, Neil Gershenfeld from the Massachusetts Institute of Technology (MIT), and Mark Kubinec from the University of California at Berkeley created the first two qubits quantum computer that could be loaded with data and its output represented a solution. Thus, using the definition of Ref. 2, a quantum computer can be defined mathematically correctly. Let $s(n)$ in be a string of classical symbols, and let the program P of size r be a symbolic representation of an algorithm.

Definition 1. An ideal quantum computer is a hypothetical device that accepts as input a classical bit string $s(n)_{in}$, and a quantum program P with size r , acting on a Hilbert space \mathcal{H}_n of dimension 2^n . For any given program P , the quantum the computer produces the classical output bit string $s(m)_{out}$ with probability as in Ref. 2

$$P_P(s_{out}^{(m)}|s_{in}^{(m)}) = P_P \langle s_{in}^{(n)} | U_P^\dagger \mathbb{I}_{n-m} \otimes (|s_{out}^{(m)}\rangle \langle s_{out}^{(m)}|) U_P |s_{in}^{(n)}\rangle$$

where $|s_{in}^{(n)}\rangle$ encodes the input string and U_P is related to the quantum program. The total amount of resources used by the device scales polynomially in r . The main differentiation property of quantum computing is the superposition state allowance which gives birth to the concept of qubit.

2.2 Quantum Computation

2.2.1 The qubit

In classical computation, the basic unit for information is the bit. Quantum computation and quantum information are built upon an analogous concept, the qubit. It is defined as a mathematical object where information can be kept as in Ref. 3 with certain specific properties. Considering them as abstract objects allows the possibility of building a theory of quantum computation and quantum information that does not depend upon a specific experimental method to obtain these qubits. It has a state on either $|0\rangle$, $|1\rangle$, or in a combination of the previous states, often called *superposition*:

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

We use Dirac notation for representing each state: $|\rangle$. Where α and $\beta \in \mathbb{C}$. We can represent their states in a two-dimensional complex vector space:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad |\psi\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$$

The states $|0\rangle$ or $|1\rangle$ become the computational basis states, an *orthonormal* basis, a set of vectors is orthonormal if it is an orthogonal set having the property that every vector has a magnitude of 1, in this vector space. Measuring α or β values separately we measure a qubit state we get either the result 0, with probability $|\alpha|^2$, or the result in 1, with probability $|\beta|^2$ where $|\alpha|^2 + |\beta|^2 = 1$. It can be noticed that the state is normalized.

Experimentally, there are different approaches to get a qubit like: nuclear magnetic resonance (NMR) as in Refs. 19, 20, atoms in optical lattices as in Refs. 21, 22, atoms in coupled cavity arrays as in Refs. 23, 24, spin lattices as in Refs. 25, 26, quantum dots as in Refs. 27, 28, and photonic systems as in Refs. 29, 30. The concept behind these experimental generation quantum information processors is that they must provide a stable ground state (see Fig. 2.1.a), an excited state (see Fig. 2.1.b), and a superposition of the previous states (see Fig. 2.1.c).

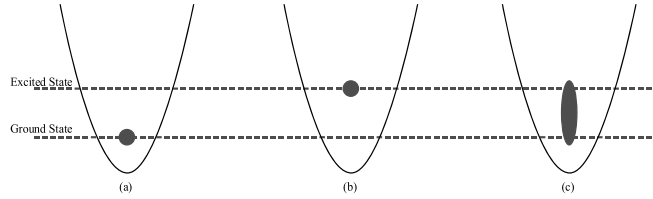


Figure 2.1: Graphical representation of a particle's energy states. (a) shows a particle in its ground state which can be associated with the $|0\rangle$ qubit state. (b) shows a particle in its excited state which can be associated with the $|1\rangle$ qubit state. (c) shows a particle in its superposition state which can be associated with the $\alpha|0\rangle + \beta|1\rangle$ qubit state.

Mathematically, a qubit (2.1) may be reformulated by using the Euler Formula as:

$$|\psi\rangle = \cos \frac{\theta}{2} |0\rangle + e^{i\varphi} \sin \frac{\theta}{2} |1\rangle \quad (2.2)$$

Where θ and $\varphi \in \mathbb{R}$. As it is shown in Ref. 3. We can visually represent the state of the qubit by using the Bloch sphere, as shown in Figure 2.2. In theory, the amount of information that can be kept in a single qubit is infinite since there are many points within it. There is no straightforward generalization of the Bloch Sphere known so far for multiple qubits.

2.2.2 System of Multiple Qubits

The mathematical space where qubits reside is the *Hilbert Space* - it is a linear vector space, finite or infinite, where its vectors have a complete inner product. A system of n qubits will have a corresponding Hilbert space of dimension $N = 2^n$. Thus, the computational basis states of the system have the form:

$$|x_1 x_2 \dots x_n\rangle = |x_1\rangle \otimes |x_2\rangle \dots \otimes |x_n\rangle,$$

Where \otimes denotes a tensor product, any linear combination of this basis is a valid state. A multiple qubit system may behave in two ways. First, the qubits measurements are independent of each other, then the state of the extensive system can be written as a tensor product and is called a separable state. On the other hand, qubits' measurements can be correlated among them, in a non-local, non-classical way, and then we say that these are entangled qubits. Clear examples of this case are the Bell state as in Ref. 31, which shows the simplest and maximal of quantum entanglement between two qubits.

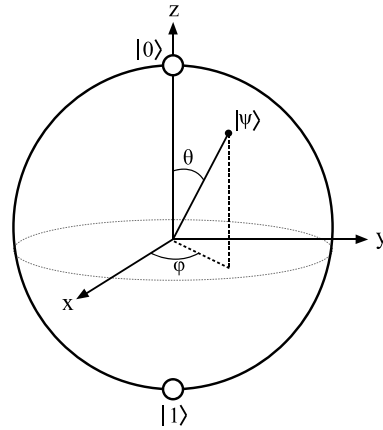


Figure 2.2: Bloch sphere representation of a qubit, where θ is the polar angle and ϕ is the azimuthal angle. Adapted from Ref. 3.

Theoretically, we can have a n – *qubit* system with no limitation; experimentally, it becomes a headache. For instance, let $n = 500$ qubits, the amount of storage required to save this amount of information is larger than the number of atoms in nature $\approx 10^{82}$, according to Ref. 32. Currently, the maximum number of ready-to-work qubits is 100, developed by IBM as Ref. 33 shows.

2.2.3 Quantum Gates

As in classical computation, quantum gates are elementary operations that alter the state of a qubit. In the computational basis, these operators are represented by matrices which are *unitary*, U is a unitary matrix if $U^H = U^{-1}$, where U^H is the *conjugated transpose matrix*, $U^H = \overline{U^T}$ and U^{-1} is the inverse matrix and $\hat{U}^\dagger \hat{U} = \hat{I}$. There are two types of quantum gates that act in a single qubit and multiple qubit gates.

Single quantum gates

One example of a single-qubit gate is the quantum NOT gate. It takes $|0\rangle$ to $|1\rangle$ state and vice versa. Matricially, it is expressed as:

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

Other single-qubit-gates are the Y and Z gates which together with X matrix are known as the Pauli matrices.

Their matrix representations are given by:

$$Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}; \quad Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad (2.3)$$

Furthermore, another essential and valuable quantum gate is the Hadamard gate. It turns $|0\rangle$ into $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, and turns $|1\rangle$ into $\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$:

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (2.4)$$

Although there exist numerous single-qubit gates, all may be decomposed into the general form as Ref. 3 shows:

$$U = e^{i\alpha} \begin{bmatrix} e^{-i\beta/2} & 0 \\ 0 & e^{i\beta/2} \end{bmatrix} \begin{bmatrix} \cos \frac{\gamma}{2} & -\sin \frac{\gamma}{2} \\ \sin \frac{\gamma}{2} & \cos \frac{\gamma}{2} \end{bmatrix} \begin{bmatrix} e^{-i\delta/2} & 0 \\ 0 & e^{i\delta/2} \end{bmatrix}, \quad (2.5)$$

where α , β , and δ are real-valued. The second matrix is just an ordinary rotation. The first and the last matrices are rotations in a different plane. This decomposition can be used to give an exact prescription for performing an arbitrary single quantum logic gate.

Multiple qubit gates

The multi-qubit quantum gates act on more than one qubit at once. For instance, the quantum CNOT gate has two qubits fixed like the control qubit and the target qubit, respectively. It reverses the state of the target qubit if and only if the control qubit is in $|1\rangle$ state as shown in the table 2.1. Its matrix representation is represented in the equation 2.6.

Before		After	
Control	Target	Control	Target
$ 0\rangle$	$ 0\rangle$	$ 0\rangle$	$ 0\rangle$
$ 0\rangle$	$ 1\rangle$	$ 0\rangle$	$ 1\rangle$
$ 1\rangle$	$ 0\rangle$	$ 1\rangle$	$ 1\rangle$
$ 1\rangle$	$ 1\rangle$	$ 1\rangle$	$ 0\rangle$

Table 2.1: Two-qubit gate Controlled NOT.

$$C_{not} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (2.6)$$

2.2.4 Measurement

It is represented by a "meter" symbol. It converts a single qubit state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ into a probabilistic classical bit M , which is 0 with probability $|\alpha|^2$, or 1 with probability $|\beta|^2$.

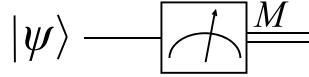


Figure 2.3: Quantum circuit symbol for measurement adapted from Ref. 3.

Fidelity of states

In this thesis, to compare the output probabilities from the different quantum processors, we will use the *classical fidelity* definition.

Definition 2. Let $p = \{p_i\}$ and $q = \{q_i\}$, where $i = 1, \dots, n$, be the probability distributions. The fidelity between p and q is defined as:

$$F'(p; q) = \sum_{i=1}^n \sqrt{p_i q_i} \quad (2.7)$$

The fidelity is bounded as $0 \leq F' \leq 1$ and it reaches the value 1 only if $p = q$. Alternatively, it can be expressed in percentage, such $0 \leq F' \leq 100$. Additionally, to analyze the measurements we will use the criteria of absolute and relative error.

Definition 3. Let x_r be the real value, and x_i the measured value. Thus the *absolute error* ϵ is given by

$$\epsilon = |x_i - x_r| \quad (2.8)$$

and the *relative error*, ϵ_r is given by

$$\epsilon_r = \frac{\epsilon}{x_r} \quad (2.9)$$

2.3 Quantum Simulation

As Deutsch-Jozsa suggested in Ref. 34, quantum computers might be capable of solving some computational problems much more efficiently than classical computers. Generally speaking, three classes of quantum algorithms help to perform better than the known classical counterparts. Although this simulation process can be executed in classical computers, they are not efficient. classical algorithms: Quantum Fourier Transforms Ref. 35, Quantum Search Algorithms Ref. 36, and Quantum Simulation Ref. 3.

Using quantum computers to simulate nature through a set of mathematical identities and its complexity is an obvious candidate. The main problem is that the number of complex numbers needed grows exponentially.

Generally, storing the quantum state of a system with n distinct components takes something like c^n bits of memory on a classical computer as Ref. 37 indicates, where c is a constant which depends on the details of the system and the desired accuracy. Conversely, a quantum computer can perform the simulations using kn qubits, where k is a constant that depends upon the simulated system's details. This makes it feasible to simulate quantum systems using quantum computers. However, it does not mean that having a fast simulation will allow the desired information about the quantum system to be obtained. When measured, a kn qubit simulation will collapse into a definite state as Ref. 38 explains, giving only kn bits of information; the c^n bits of "hidden information" in the wavefunction are not entirely accessible.

The heart of simulations is the solution of differential equations that capture the physical laws governing the system's dynamical behavior. The objective is that given an initial state of the system, we must predict what will be the state at some other time and position. Therefore, it is necessary to approximate the solution and then discretize the differential equation in space and time. An iterative application of a procedure carries the initial state to the final conditions. As you may be guessing, the error through this process can grow fast; this is why it is bounded into a δ maximum.

Let us start by exploring the physics behind the simulation of these quantum systems. Schrodinger's equation describes the behavior of most dynamical quantum system, except relativistic dynamical systems:

$$i\hbar \frac{\partial}{\partial t} |\psi(t)\rangle = \hat{H} |\psi(t)\rangle \quad (2.10)$$

A typical Hamiltonian of interest to physicists, if we were just dealing with a real single particle in a potential and stationary solutions, makes 2.10 be reduced to:

$$i\hbar \frac{\partial}{\partial t} \psi(x) = \left[-\frac{\hbar^2}{2m} \frac{\partial^2}{\partial x^2} + V(x) \right] \psi(x) \quad (2.11)$$

Importantly, the interactions of multiple particles can be modeled in a similar way, using the split-step method as in Ref. 39. However, for the sake of clarity, we focus herein on Hamiltonians of the form given by Eq.(2.11). The critical challenge to simulating these quantum systems is the exponential differential equations that must be solved. For instance, two differential equations must be solved for one qubit evolving according to (2.11). For two qubits, it grows to 4, and for n qubits, 2^n equations. Thus, simulating these systems with quantum computers may allow us to explore interest quantum systems for which classical computers would not.

In a quantum simulator, the evolution of the physical model is *mapped*, following the rule of quantum mechanics, onto the practical algebra of quantum registers made of qubits. The *quantum time propagator*, namely a unitary operator, can be programmed in digital steps using a sequence of quantum logic states (unitary transformations on the qubits) defining a quantum circuit as Ref. 3 states.

The Schrodinger equation (2.10) can be fully solved in principle by computing the unitary time-evolution operator $\hat{U}(t) = e^{-i\hat{H}t}$. Once it is known, any initial condition can be evolved linearly as:

$$|\psi(t)\rangle = \hat{U}(t) |\psi(0)\rangle \quad (2.12)$$

In the following paragraphs, we will review some recipes both for mapping the target system and for the translation of evolution operators into quantum register operations.

2.3.1 From Feynman to Lloyd

In 1982, Feynman conjectured that using a controllable quantum mechanical system as a computing resource would allow simulating quantum systems with better performance than a classical computer. However, it was not until 1996 that Seth Lloyd proved this idea, see Ref. 40, with a single limitation that the systems to be simulated can carry local interactions between their constituent subsystems. Thus, we will concentrate on system Hamiltonians of the form:

$$\hat{H} = \sum_{k=1}^L \hat{H}_k \quad (2.13)$$

where \hat{H}_k are often near terms interactions such as $X_i X_j$. Additionally, there are three symmetry constraints such as particle statistics. Thus, it is easier to simulate (2.13) than (2.11) because it acts into a subsystems, the next step is to implement using a quantum circuit. Since $[\hat{H}_j, \hat{H}_k] \neq 0$, in general $e^{-i\hat{H}t} \neq \prod_k e^{-i\hat{H}_k t}$. Therefore, a question arises, how can we use (2.13) to construct (2.11)? We shall take advantage of the Suzuki-Trotter formula stated in Ref. 41, the heart of quantum simulation algorithms.

Theorem 1. Let \hat{A} and \hat{B} be Hermitian operators. Then

$$e^{\hat{A}+\hat{B}} = \lim_{n \rightarrow \infty} (e^{\hat{A}/n} e^{\hat{B}/n})^n$$

Where its first order expansion represents:

$$e^{(\hat{A}+\hat{B})\Delta t} \approx (e^{\hat{A}\frac{\Delta t}{n}} e^{\hat{B}\frac{\Delta t}{n}})^n + O\left(\frac{\Delta t^2}{n}\right) \quad (2.14)$$

where Δt is the time-steps size where if $\Delta t \ll 1$ the O approximation is negligible; n is the total number of time-steps. Note that Theorem 1 is true regardless the commutative properties of A and B operators, see Ref. 42. Following the Suzuki-Trotter Formula, let's focus in the physical simulation where $\hat{H} = \hat{K} + \hat{V} = \frac{\hat{p}^2}{2m} + V(\hat{x})$, where p is the momentum operator and x is the position operator. Thus, the time evolution operator is given by:

$$\hat{U}(\Delta t) = e^{-i\hat{H}\Delta t} = e^{-i\hat{K}\Delta t} e^{-i\hat{V}\Delta t} + O(\Delta t^2) \quad (2.15)$$

Thus, (2.12) becomes:

$$|\psi(t)\rangle \approx e^{-i\hat{K}t} e^{-i\hat{V}t} |\psi(0)\rangle \quad (2.16)$$

2.3.2 Universality of the Hamiltonian's Simulation

Using (2.13), we may describe a wide range of different potentials representing different physical systems by only varying the potential term. Both Hamiltonian components, kinetic and potential operators, are expressed as functions

of momentum and position operators, respectively. We have that operators represented by diagonal unitaries of the form $e^{if(x)}$ can be implemented generically by using a correspondence between Walsh functions stated in Ref. 43 and a basis for diagonal operators following Welch's method without using an ancilla qubit. Therefore, let us take a look at the foundations of Walsh Functions.

Walsh Functions

The Paley-ordered Walsh functions are defined on the continuous interval $0 \leq x < 1$, see Ref. 44:

$$w_j(x) = (-1)^{\sum_{i=1}^n j_i x_i} \quad (2.17)$$

where $j \in \mathbb{Z}^+$. Walsh functions form a *complete* and *orthonormal* set. Here j_i is the i th bit in the binary expansion, $j = \sum_{i=1}^n j_i 2^{i-1}$. Thus, $j = (j_n, j_{n-1}, \dots, j_1)$ and the n th term is the index of the most significant non-zero bit; and x_i is the i th bit in the dyadic expansion, $x = \sum_{i=1}^{\infty} \frac{x_i}{2^i}$ where $x = (x_1, x_2, \dots, x_n)$ is the dyadic representation of x , the most significant bit is on the left. In Figure 2.4, we can see the first 4 Walsh functions.

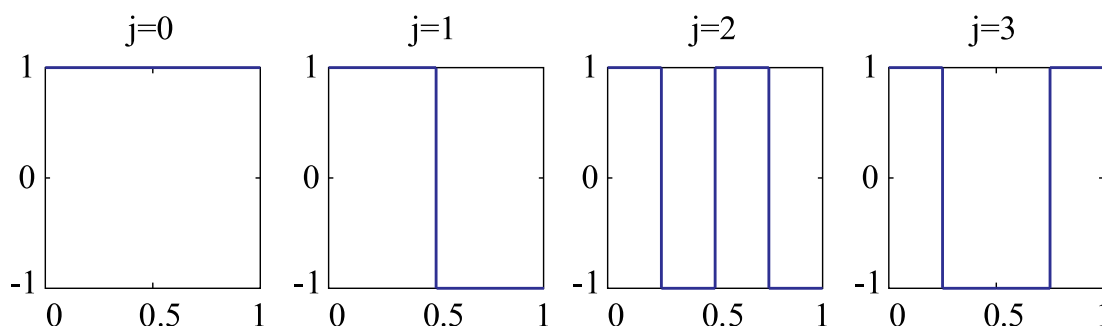


Figure 2.4: First four Walsh functions, in Paley order. adapted from Ref. 43.

Walsh functions can be used as a basis for orthonormal expansion. If we are treating with a discrete function, it can be performed by a discrete Walsh-Fourier transform. For arbitrary n , let us discretize the interval $[0, 1)$ into $N = 2^n$ points, $x_k = k/N$, where $k = [0, N - 1]$. Let's define discrete Walsh functions as $w_{jk} = w_j(x_k)$. In terms of the bits of j , k , and x , we have:

$$w_{jk}(x) = (-1)^{\sum_{i=1}^n j_i x_i} \quad (2.18)$$

where k_i is the i th bit in the dyadic expansion as before. (2.18) shows whether x is continuous or discrete, the only difference will be the number of bits in the expansion of x . This makes the Walsh series useful for representing discretely sampled functions. In the discrete case, orthonormality and completeness properties are obtained. Thus, the discrete Walsh-Fourier transform a_j of a function $f_k = f(x_k)$ is:

$$a_j = \frac{1}{N} \sum_{k=0}^{N-1} f_k w_{jk} \quad (2.19)$$

$$f_k = \sum_{j=0}^{N-1} a_j w_{jk} \quad (2.20)$$

For expansions in trigonometric functions, the relevant group is that of translations. For Walsh functions up to order 2^n , it is the group $Z^{\otimes n}$ which is formed by a basis for diagonal operators on n qubits; these operators are called the Walsh operators, which will be introduced below.

Walsh Operators

The state of an n -qubit register in a quantum computer is expressed as a superposition:

$$|\psi\rangle = \sum_{k=0}^{N-1} c_k |k\rangle \quad (2.21)$$

of $N = 2^n$ states in the computational basis, see Ref. 43, defined as $|k\rangle = |k_1, \dots, k_n\rangle$. $k = 0, 1, \dots, N-1$ is represented as an n -bit dyadic expansion, as in Walsh series. The bits $k_i = 0$ or 1 denote the state of the i th qubit. A unitary operator $U = e^{if(x)}$ that is diagonal in this basis is given in terms of its eigenvalues as $f|k\rangle = f_k|k\rangle$.

Functions $f(x)$ of a continuous variable $x \in [0, L)$ may be represented following this procedure if they are discrete. For instance, let $L = 1$, and the (grid) points as $x_k = kL/N$, so that $f_k \equiv f(x_k)$. Thus $x_k \in [0, 1)$. So, let \hat{Z}_i denote the Pauli \hat{Z} operator acting on the i th qubit:

$$\hat{Z}_i |k_1, \dots, k_n\rangle = (-1)^{k_i} |k_1, \dots, k_n\rangle$$

We define the Walsh operator of order j on n qubits as:

$$\hat{w}_j = \otimes_{i=1}^n (\hat{Z}_i)^{j_i} = (\hat{Z}_1)^{j_1} \otimes (\hat{Z}_2)^{j_2} \otimes \dots \otimes (\hat{Z}_n)^{j_n} \quad (2.22)$$

where $j = 1, \dots, 2^n$, and j_i is the i th bit in the binary expression. Powers of $(\hat{Z}_i)^1 = \hat{Z}_i$ and $(\hat{Z}_i)^0 = 1$. The set of all Walsh operator $j = 1, \dots, 2^n$ forms a basis for diagonal operator on n qubits, given by all possible tensor products of single-qubit Pauli \hat{Z} gates. Their eigenvalues in the computational basis $|x\rangle$, $x \in [0, 1)$, are Walsh functions index j and independent variable x :

$$\hat{x}_j |x\rangle : \otimes_{i=1}^n (\hat{Z}_i)^{j_i} |k\rangle = \prod_{i=1}^n (-1)^{j_i k_i} |k_1, \dots, k_n\rangle = w_{jk} |k\rangle = \hat{w}_j(x) |x\rangle \quad (2.23)$$

Thus, the locations of the \hat{Z} operators in \hat{w}_j correspond to the position of the 1's in the *bit reversed* binary string for j . For instance: The Walsh operator, w_{ij} , where $j = 6$ on $n = 3$ qubits is $w_6 = 1 \otimes \hat{Z} \otimes \hat{Z}$ since $j = 6$ in binary is $(j_3 j_2 j_1) = (110)$. The gate representation of w_6 is shown in Figure 2.5. The general Walsh operator requires $O(n)$ gates for its implementation a single Z gate and up to 2^n controlled NOTs.

Using (2.20), any diagonal operator on n qubits may be expanded as a sum of $N = 2^n$ Walsh operators, $f = \sum_{j=0}^{N-1} a_j w_j$ since these operators commute characteristic. Therefore, any diagonal unitary may be written as a product of exponential of Walsh operators:

$$\hat{U} = e^{if} = \prod_{j=0}^{N-1} e^{ia_j \hat{w}_j} \quad (2.24)$$

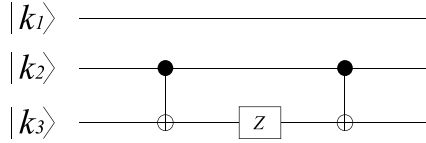


Figure 2.5: w_6 gates implementation adapted from Ref. 43.

Each term of the product of $\hat{U}_j = e^{ia_j \hat{w}_j}$ has the form $\exp(-i\frac{\theta_j}{2} \otimes_i (Z_i)^{j_i})$, where $\theta_j = -2a_j$. Hence, it is similar to (2.22), with only one difference: the Z-gate is replaced by a \hat{Z} -rotation, $\hat{R}_z(-2a_j)$, where $\hat{R}_z(\theta) \equiv e^{-i\hat{Z}\theta/2}$. Thus, the circuit for \hat{U} is given by successively applying the circuits for \hat{U}_j .

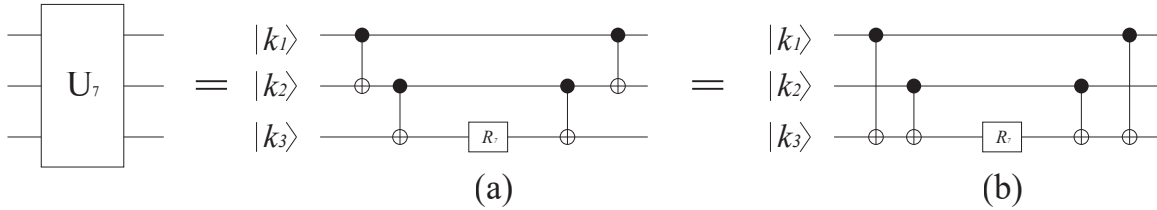


Figure 2.6: w_7 gates implementation using $n = 3$ qubits, equivalent for implementing the operator $U_7 = \exp(ia_7(\hat{Z} \otimes \hat{Z} \otimes \hat{Z}))$. (a) shows one possible implementation, but (b) shows our preferred implementation method where the CNOTs are always targeted on the highest order qubit possible adapted from 43.

Figure 2.6 shows two ways of implementing U_7 . As we see, there is not a unique gate configuration. Therefore, we are going to use the convention of Welch using Ref. 43 where the CNOTs are always targeted on the highest order qubit possible. Then a rule for constructing the circuit U_j can be given in terms of the binary expansion of j :

A rotation gate, $\hat{R}_z(-2a_j)$, is placed on the qubit corresponding to the most significant non-zero bit (MSB) of j . Then CNOTs are placed on either side, targeted on the same qubit as the rotation gate, and controlled on the qubits corresponding to the 1's other than the MSB in the binary expansion of j .

(2.24) can be generalized for more than one dimension easily. For a d -dimensional system represented by d registers of n qubits each, the single Walsh operators will be replaced by tensor products of up to d Walsh operators over the different registers. The actual number depends on the number of variables in the function f . It does not increase the gate complexity as interaction potentials are generally few-body, and the product of Walsh operator are also other Walsh operators. Additionally, this representation allows us to a reduction in the circuit depth for implementing U if some of these coefficients are zero, see Ref. 43.

2.3.3 Quantum Fourier Transform

It acts on a quantum state $|X\rangle = \sum_{j=0}^{N-1} x_j |j\rangle$ and maps it to the quantum state $|Y\rangle = \sum_{k=0}^{N-1} y_k |k\rangle$ according to the formula:

$$y_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} x_j w_N^{jk} \quad (2.25)$$

where $w_N^{jk} = e^{2\pi i \frac{jk}{N}}$. It can be represented by the unitary matrix:

$$\hat{U}_{QFT} = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} w_N^{jk} |k\rangle \langle j| \quad (2.26)$$

The Inverse Quantum Fourier Transform may be obtained by taking the inverse of (2.26) since they are unitary and hermitian matrices.

2.3.4 Initialization

To simulate the wave function, we must provide an initial state which can be a delta function, gaussian wave, or a sinusoidal wave. Thus, it may be mapped using the Walsh functions, and implemented by using the Walsh Operators into Z -rotations. in this thesis we will use a delta function which will be represented as an X -gate.

2.4 IBM Quantum Computers

ust like classical computers, quantum computers need to have some internal and external physical conditions to work properly, such as:

1. **A suitable temperature.** In order for an IBM Quantum Computer works appropriately, the quantum processor must remain cold - about milikelvins above the absolute zero. IBM uses super-cooled superfluids to create superconductors.
2. **A suitable platform.** At ultra-low temperatures, the processor's material becomes superconducting, and electrons move through them without resistance forming "Cooper pairs." These pairs produce quantum tunneling, carry a charge across barriers—two superconductors placed on either side of an insulator form a Josephson junction, which together with a capacitor form a transmon qubits ? .
3. **A suitable control system.** IBM quantum computers use transmon qubits, with a particular configuration operating a microwave regime, as superconducting qubits. Therefore, by firing microwave photons at these qubits, they can control, change, and read out the qubits information.
4. **A set of suitable properties.** In order to create a multidimensional computational space, qubits are placed in a *superposition* state. Thus, complex problems can be represented in new ways in these spaces. Additionally, these qubits can be *entangled* in order to explore this quantum mechanical feature.

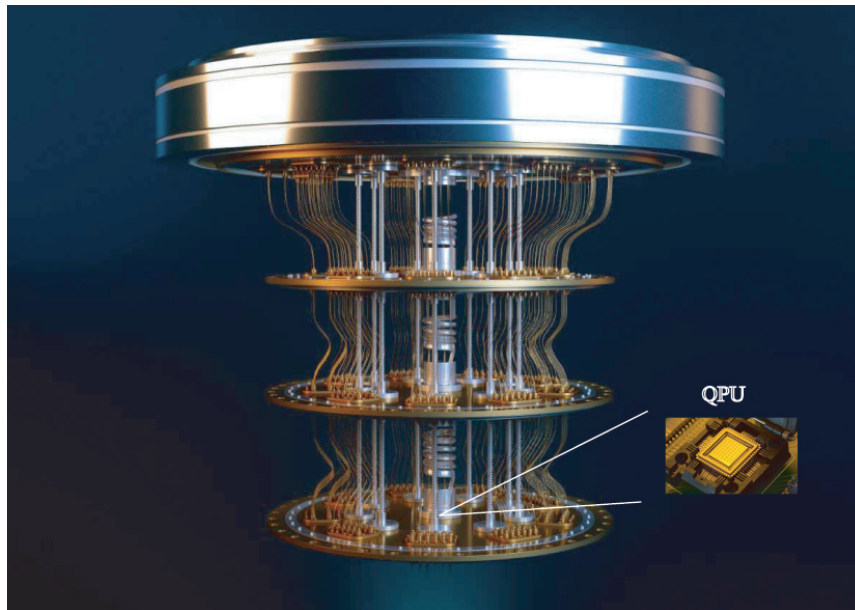


Figure 2.7: Quantum Computer Hardware, courtesy from IBM.

IBM-Q has created the QISKIT (Quantum Information Software Kit) project, which supports *Python3.6* or later 15; it is highly recommended to use *Anaconda* as a cross-platform Python distribution for scientific computing. The workflow used in Qiskit consists of three high-level steps:

- **Build:** design a quantum circuit that represents the problem you are considering
- **Execute:** run experiments on different backends (quantum computers and simulators)
- **Analyze:** calculate summary statistics and visualize the results of experiments.

Qiskit has four computing elements where specific tasks are executed. *Terra* composes quantum programs at the level of circuits, optimize them for the constraints of a particular device, and manages the execution of batches of experiments on remote-access devices. *Aer* is used to understand the limits of classical processors by demonstrating to what extent they can mimic quantum computation. *Ignis* is focused on fighting noise and errors and forging a new path like errors characterization, improving gates, and computing in the presence of noise. *Aqua* is focused on finding out real-world applications. This is the place where quantum algorithms are designed and built.

IBM-Q has 23 quantum computers available in its quantum computing platform. Nonetheless, in this thesis, we will use *ibmq – quito* and *ibmq – lima* to perform our simulations. Their performance is evaluated by a function of their technical characteristics, and architecture which are found in Table 2.2, we have the next terms:

1. **Quantum Volume.** According to Nikolaj Moll et. al in Ref. 45, It depends on the number of qubits N as well

as the number of steps that can be executed, the circuit depth:

$$\widetilde{V}_Q = \min[N, d(N)]^2$$

However, IBM's researchers modified its definition to be an exponential of the circuit size:

$$\log_2 V_Q = \max_{n \leq N} \{ \min[n, d(n)]^2 \}$$

2. **CLOP** It is a metric correlated with how fast a quantum processor can execute circuits as in Ref. 15.
3. **Transpilation** It is a process of rewriting an input circuit to match the topology of an specific quantum device, and optimize their time execution by using the basis gates associated with each quantum computer.

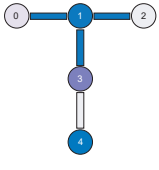
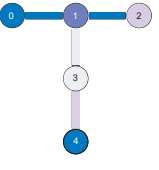
Backend	ibmq-quito	ibmq-lima	
Number of Qubits	5	5	
Quantum Volume	16	8	
CLOP	2.5	2.7	
Processor	Falconr4T	Falconr4T	
Version	1.1.26	1.0.33	
Basis Gates	CX, ID, RZ, SX, X	CX, ID, RZ, SX, X	
Gate time (ns)	295.111	405.333	

Table 2.2: Technical Characteristics of IBM-Q processors, adapted from the IBM Quantum Experience, adapted from Ref. 1

Chapter 3

Methodology

3.1 Algorithm Development

3.1.1 Time Evolution Algorithm Design

The Universal Quantum Simulator seeks for generating a quantum gate circuit implementation, automatically, of physical systems which are represented like (2.13) which allows us to implement a modular-building logic behind each quantum system. Once we have a physical quantum system represented using a Hamiltonian representation, we can simulate the system by letting it evolve from the initial state like (2.12) using the real-space, or first-quantized, representation of the wave-function in terms of position eigenstates as:

$$|\psi(t)\rangle = \int |x\rangle \langle x| |\psi(t)\rangle dx \quad (3.1)$$

Each x^i is discretized and represented on the quantum computer in the computational basis as in (2.19) of n qubits each; the basis states corresponding to a grid of 2^{dn} bits. We can perform this change of basis independently since this change of basis works for all diagonal unitary matrices, see Ref. 43. Thus, they have the form (2.24). For our calibration, we will use a pulse represented by an X -gate in an arbitrary qubit as in Figure 3.4. (2.12) is evaluated by using the Trotterization formula represented by the theorem (1), which is true regardless of the commutativity of their operators. Since we have a kinetic and potential Hamiltonian terms like (2.15), (2.12) becomes (2.16) where Δt is an integer called the Trotter number. The next step is to diagonalize the kinetic operator \hat{K} by using the Quantum Fourier Transform (QFT) and its inverse (IQFT), which are represented by F and F^\dagger operators as in Figure 3.1, and Figure 3.2, respectively.

The final \hat{K} operator circuit implementation is shown in Figure 3.3. We do not need to diagonalize the potential operator \hat{V} as it is already expressed into the position x representation. The final quantum algorithm that simulates (2.12) is, and its implementation is shown in Figure (3.4) :

$$|\psi(t)\rangle = (\hat{F}^\dagger e^{-i\hat{K}\Delta t} \hat{F} e^{-i\hat{V}\Delta t})^{\frac{t}{\Delta t}} |\psi(0)\rangle \quad (3.2)$$

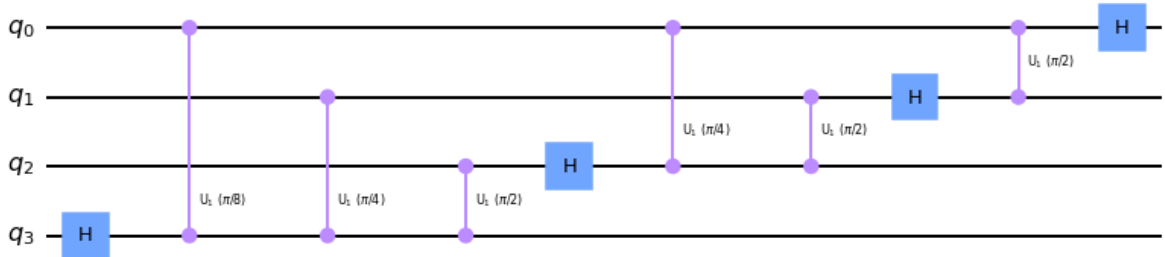


Figure 3.1: 4 – qubit Quantum Fourier Transform

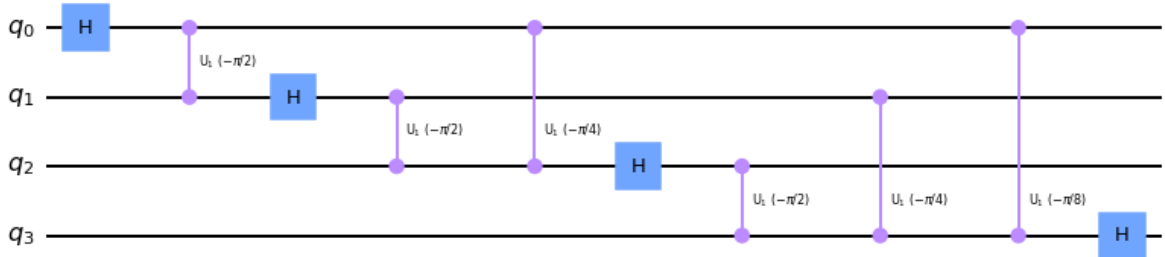


Figure 3.2: 4 – qubit Inverse Quantum Fourier Transform

The process how we will proceed on executing the Universal Quantum Simulator will be following two main stages:

- **Statevector Simulator.** Here we will use a noiseless simulator to perform our simulation. It will give us a set of probabilities distributions along time evolution. Additionally, if we place the measurement component we will let the qubit be collapsed in a certain state based on its probability distribution.
- **IBM-Q Experience.** After our noiseless simulations, we can outperform the same circuit implementation on quantum computers by using the IBM terminal. There are two main characteristics associated to this stage, by one side the circuit will be transpiled, re-written, in function of the basis gates of each processor; by another side, the processor will send a series of electrical signals called shots which result states will be averaged and its average is the final probability will be displayed as the final result.

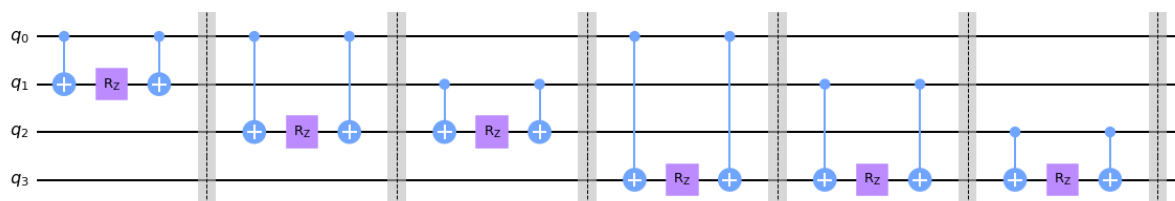


Figure 3.3: Momentum Gate Implementation

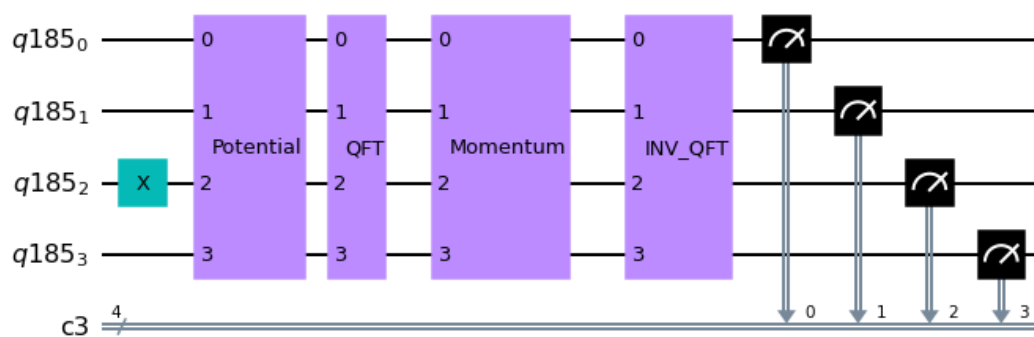


Figure 3.4: Circuit Gate Modular Implementation

Chapter 4

Results & Discussion

4.1 Calibration

We use a 1D harmonic oscillator as the calibration physical system for the Universal Quantum Simulator's variables. Thus, let us consider a potential given by:

$$V(\hat{x}) = a(x - x_0)^2 \quad (4.1)$$

where $a = \frac{1}{2}k$, k is the Hooke's constant given in $[N/fm]$ as well as a , x_0 represents the rest position given in $[fm]$. We will expand (4.1) by using the Walsh Functions. Thus, we set $a = 5 [N/fm]$, and $x_0 = 15 [fm]$ with $\Delta t = 0.01 [ns]$, and varying the number of used qubits for the expansion. Note, that the scale of physical units have been selected for our simulations. The absolute errors of the potential operator approximation are distributed as shown in Figure 4.1a, and for the momentum, in the Figure 4.1b

For each qubit, the absolute errors of the potential and kinetic operators were calculated in to get the *optimized* number of qubits that gives the least absolute error approximation. Table 4.1 shows the different potential and kinetic absolute errors distributions in function of the number of qubits.

As the deviation is fluctuating since the number of outliers as it can be seen in Table 4.1, it is important to select the number of qubits with the least absolute error, dispersion, and lower outliers points from Figure 4.1, to get this task, we use the relative error definition as it is shown in Figure 4.2 Thus, we use five qubits to approximate (4.1) into the Walsh Functions basis. The potential and kinetic operators are plotted in Figure 4.3a, and Figure 4.3b, respectively. As the operators expand their position, the time factor does not affect the error propagation. Thus, the error distribution of the expanded potential and kinetic operators remains the same.

Even though the absolute minimum errors were obtained when the approximation was made using six qubits (see Table 4.1), it was not chosen as the *optimized* number of qubits since the number of outliers points that it had. These outliers points are present because of the x -discretization into Walsh Functions. Furthermore, it was seen that from six qubits approximation, there are oscillations in the approximation errors, as it is seen in annex Figure (C.1). Indeed, to have a relative comparison term, we have got the relative errors from the Walsh Function approximations.

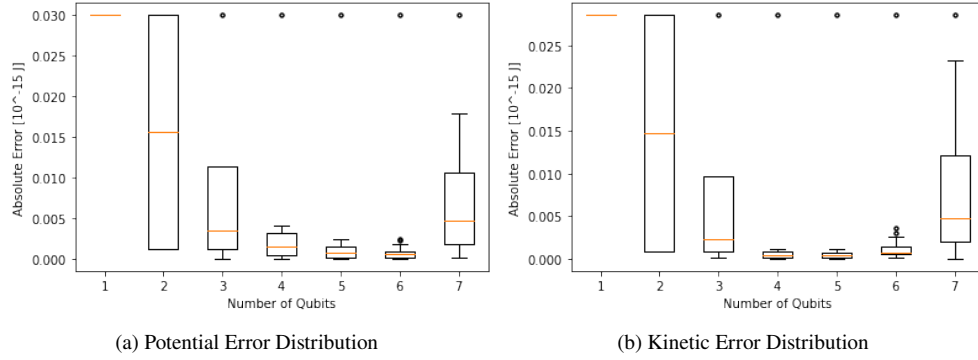


Figure 4.1: Operators Approximation with $\Delta t = 0.01$ [ns] using the Walsh decomposition.

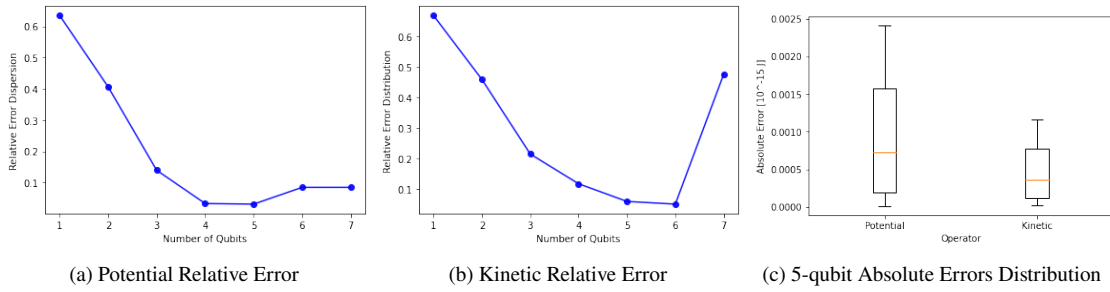


Figure 4.2: Potential and Kinetic Operators approximated using the Walsh decomposition.

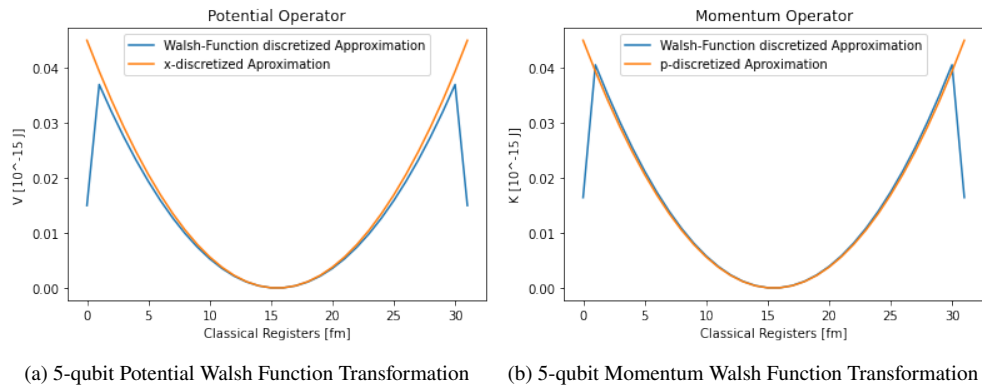


Figure 4.3: Potential and Kinetic Operators approximated in Walsh Function Basis

Number of Qubits	Potential Operator		Kinetic Operator	
	Absolute Error [$10^{-15} J$]	σ	Absolute Error [$10^{-15} J$]	σ
1	0.03	0	28.55×10^{-3}	0
2	15.62×10^{-3}	14.37×10^{-3}	14.72×10^{-3}	13.83×10^{-3}
3	3.42×10^{-3}	12.14×10^{-3}	2.25×10^{-3}	11.7×10^{-3}
4	1.52×10^{-3}	9.51×10^{-3}	4.06×10^{-4}	9.31×10^{-3}
5	7.26×10^{-4}	7.10×10^{-3}	3.67×10^{-4}	6.82×10^{-3}
6	5.55×10^{-4}	5.14×10^{-3}	7.73×10^{-4}	4.85×10^{-3}
7	4.65×10^{-3}	6.39×10^{-3}	4.74×10^{-3}	7.04×10^{-3}

Table 4.1: Operator's Error Approximation

As a result, when using five qubits, we have a relative error of 0.06 and 0.03 in the potential and kinetic operators. Thus, we use five qubits as the *optimized* number of qubits, as it can be seen in Figure 4.2, to expand our operators and to simulate our physical system.

4.2 Circuit Implementation

We will use a delta function to initialize the simulation, which is placed into the last qubit (four qubit) as a *bit-flip* (X-gate) as it is shown in Figure 4.4. This places the initial position of a particle in the middle of our coordinate system. Note that the potential, QFT, Momentum, and IQFT circuits implementations are attached in the appendix in Figure 4.16.

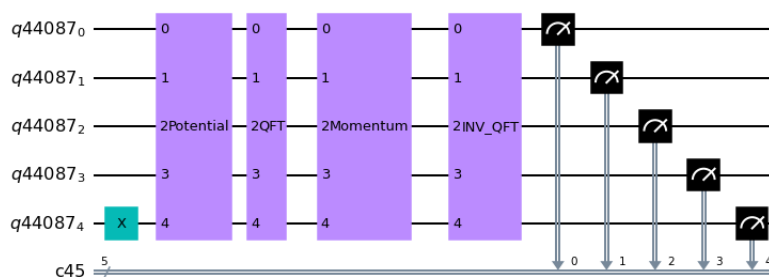


Figure 4.4: 4th-qubit position Circuit Gate Implementation of the initial position of a particle.

Once all this process is set, it is time to run in the qiskit simulator with $\Delta t = 0.01 [ns]$, and for a period of $t = 20 [ns]$. We use the *statevector – simulator* to run all simulations. This simulator gives us a noiseless result. Thus, our physical system evolves, as Figure 4.5 shows. Nonetheless, if we want an accelerated evolution, we could set a bigger Δt which will allow us to see with a broader perspective the system evolution. Thus, with this objective, we

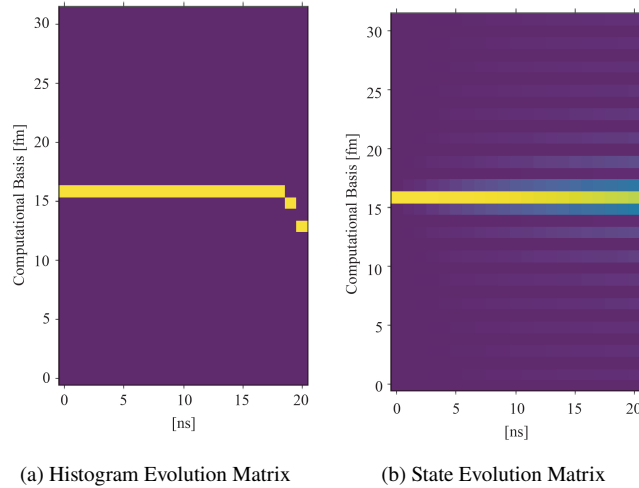
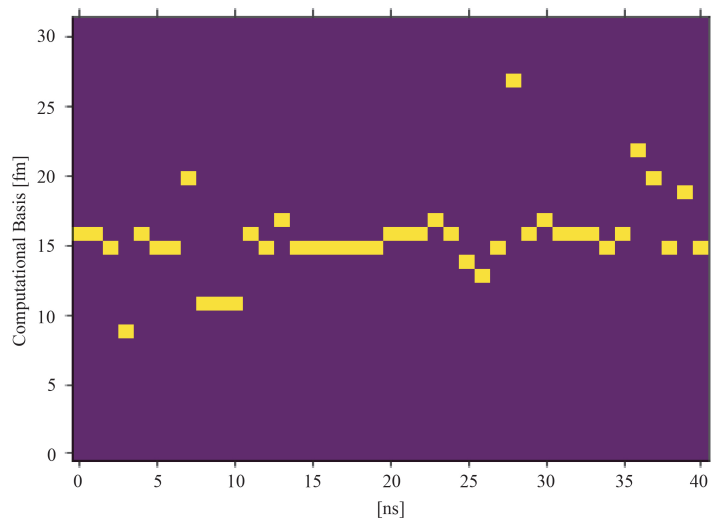


Figure 4.5: Quantum Harmonic Oscillator Evolution in the Statevector Simulator with $\Delta t = 0.01$ [ns]

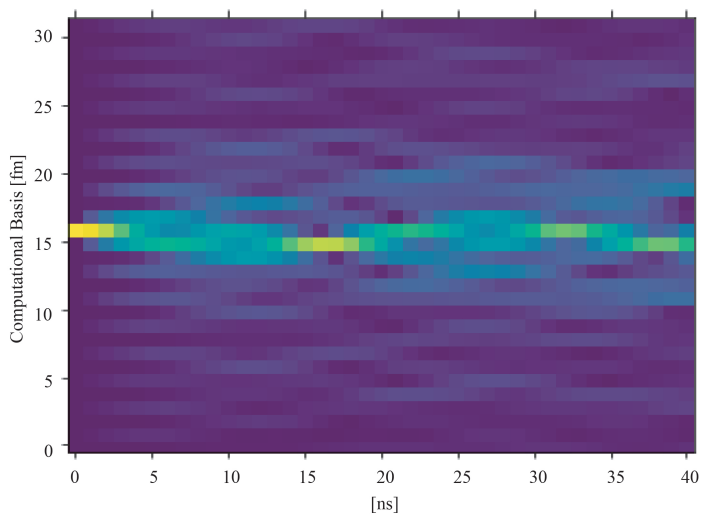
will use $\Delta t = 0.1$ [ns] and let it evolve for a larger time period $t = 40$ [ns] as it is shown in Figure 4.6.

If we compare Figures 4.5 and 4.6, We can observe a broader perspective of the system evolution, Figure 4.5 represents the two first steps of Figure 4.6 of the Harmonic Oscillator Evolution in time. The histogram evolution matrix representation results from a projective measurement of the qubit. Thus, we are affecting the qubit's state position, causing a wavefunction collapse in a state within the probabilistic distribution. On the other hand, the state vector evolution matrix represents how the states are distributed along with position (classical registers represented on the y-axis) and time evolution (x-axis) without measuring. By using the latter matrix representations, we can obtain the state probability distribution along time by using (2.1). Thus, we will measure the $|\alpha|^2$ value, which represents the probability of being at its respective state.

In Figure 4.6a we can notice how the initial state is changing as time goes on as in classical models. Its state measurement has been collapsed, so it means that there is a probability of the delta wave function is placed there as well as in other states in superposition, as it is shown in Figure 4.6a, where the initial wave is diffracted in a set of states in superposition as it is observed in known simulations like the ones represented with *Matlab*. This model was developed by using the model expressed in annex in Figure 4.7. Thus, the first two time steps of Figure 4.6a represent the ground state of the system as in Figure 4.7a, the next two time steps represent our initial state in its first excited state as in Figure 4.7b. Finally, the rest of the time evolution shows a linear combination of previous states as in Figure 4.7c. These states have energy associated with them which causes a shifting in their position over time. This evolution result was hoped as it backs up the working and usefulness state of the Universal Quantum Simulator. It is important to notice that this physical system has being widely studied computationally and this is the main reason why it was chosen, to have a well-known sand to make comparisons regrading the feasibility of our UQS. Compare the Matlab simulations of the system. We can notice that there is some blurring in the superposition



(a) Histogram Evolution Matrix



(a) Statevector Evolution Matrix

Figure 4.6: Quantum Harmonic Oscillator Evolution in the Statevector Simulator with $\Delta t = 0.1$ [ns]

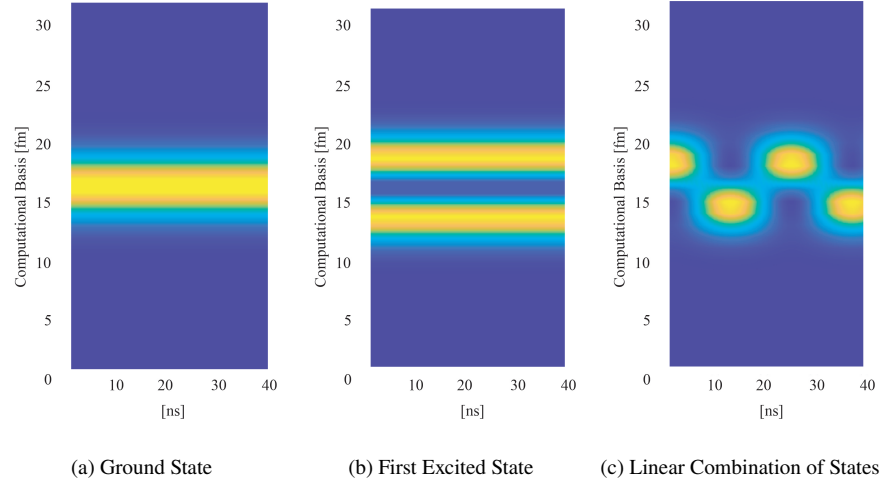


Figure 4.7: Quantum Harmonic Model using Matlab

of states because there are some phase-flips that occur during the circuit implementation, especially in the Quantum Fourier Transform. It means that there are some perturbations of the system during long-time evolution.

Letting the Quantum Harmonic Oscillator circuit system evolve during $t = 20$ [ns], $\Delta t = 0.1$ [ns] in ibmq-lima and ibmq-quito quantum computers. The evolution of the system is shown in Figure 4.8. In order to compare the performance of previously mentioned real quantum machines, we will compare their states' probabilities with the noiseless simulation developed in the previous section. Thus, we will get a state's fidelity along the time evolution. For this purpose, we will use the classical fidelity definition as shown in (2.7).

In Figure 4.8, we can see how the states' probabilities evolve. For example, the first two time steps can be shown the ground state position, the next time steps simulate the initial state in its first excited state, and as time evolves, we can notice how the superposition of both states is created. Specifically, in the 4th column, we notice how a bit-flip was produced, denoting the system evolution.

Analyzing the ibmq-lima and ibmq-quito final states' probabilities, it is vital to notice how after the first time step, the probabilities distribution degenerates along with the computational states, which is reflected in the fidelity fluctuation along time evolution shown in Table 4.2 and plot in Figure 4.9. We may suggest that as ibmq-lima and ibmq-quito have the same architecture, shown in Table 2.2 and the transpiling working principle using the same basis gates, their performance will be similar. It is partially true as although they computationally have the same running circuit, their hardware characteristics (CLOP, Processor, Gate time [ns]) affect the final measurements directly. This is the reason why the fidelity of ibmq-quito and ibmq-lima differs. Nonetheless, both fidelities fluctuations behave like an inverse parabolic function per cycle, having a maximum fidelity of 52.83 at $t = 12$ [ns] in ibmq-lima and fidelity of 52.82 at $t = 10$ [ns] in ibmq-quito. It suggests that the ibmq-quito processor evolved faster, which makes sense as this processor has a higher *quantum volume* (16); consequently, its performance is higher.

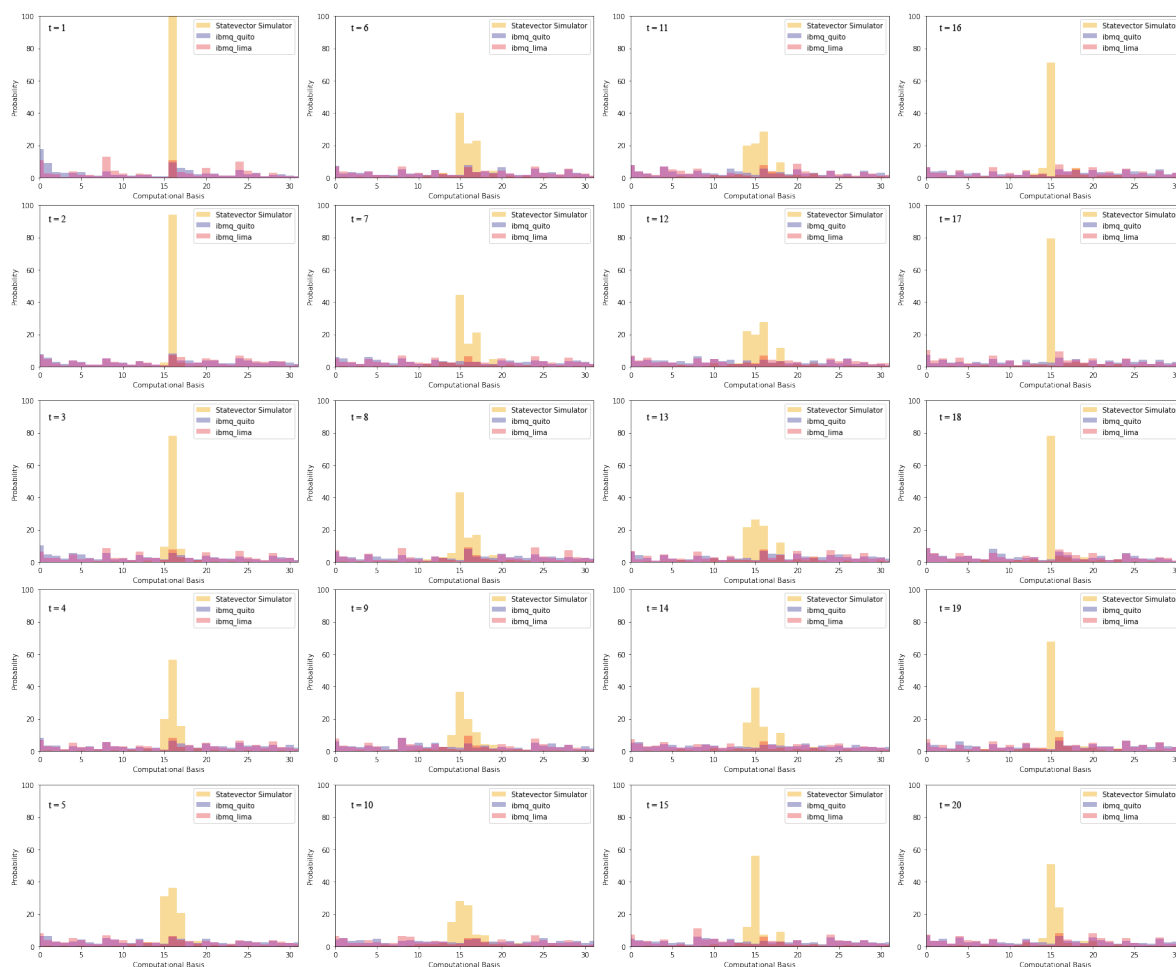


Figure 4.8: Harmonic Oscillator evolution state’s probabilities by using $\Delta t = 0.1$, $t = 20$, and 1000 shots; executed on ibmq-quito and ibmq-lima

Regarding the computational performance of the Quantum Harmonic Oscillator circuit while running in quantum processors. As both quantum computers have the same transpiling working principle and the same basis gates, both share the same *depth*, *size*, and *width* characteristics. As was mentioned in the theory chapter, the main advantage of quantum computing is that its algorithms run on time that grows linearly. Thus, the *depth* of the circuit shows this parameter. As it can be shown in Figure (4.10), QHO circuit time running evolves linearly. It suggests to us that the UQS is capable of implementing quantum circuits that can be run in some time that evolves linearly.

Finally, we can compare this QHO system evolution with other previous work like 46 where a quantum simulation of a discretized harmonic oscillator is developed. Although the UQS uses a different implementation process the

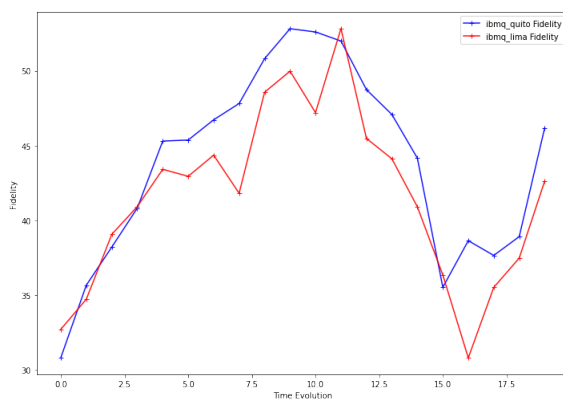


Figure 4.9: Fidelity along Time Evolution

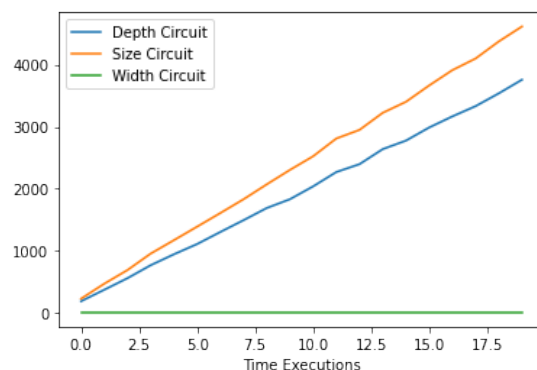


Figure 4.10: Harmonic Oscillator Performance Indicators

results obtained from quantum computers are noisy, and with a little information to extract. Our work is no exception.

4.3 Double Well Potential

Let us consider another potential, the double-well potential where its tunneling dynamics are one of the most active research areas. For example, they appeared in the mean-fields dynamics of Bose-Einstein condensates, the recent development of ion trap technology, the ultra cold trapped atoms theory, and other applications as in Ref 47. This potential is given by:

$$V(x) = E_0(ax^4 - x^2) \quad (4.2)$$

where E_0 is the ground state energy given in *hartrees*, and a is the damping factor of the double-well potential. The physics behind this physical system is quite rich, especially when $a = 0$ and $a \leq 0$. Nonetheless, in this case, we will set $E_0 = 1$ hartrees and $a = 0.5$, and we will be treated with the symmetric double-well potential. We will use

five qubits to expand our potential operator and implement our quantum circuit as in the calibration process. Thus, the potential Walsh Function approximation is shown in Figure 4.11a, and their error distribution is shown in Figure 4.12b. The median absolute error is given by 0.0066 with a $\sigma = 0.109$.

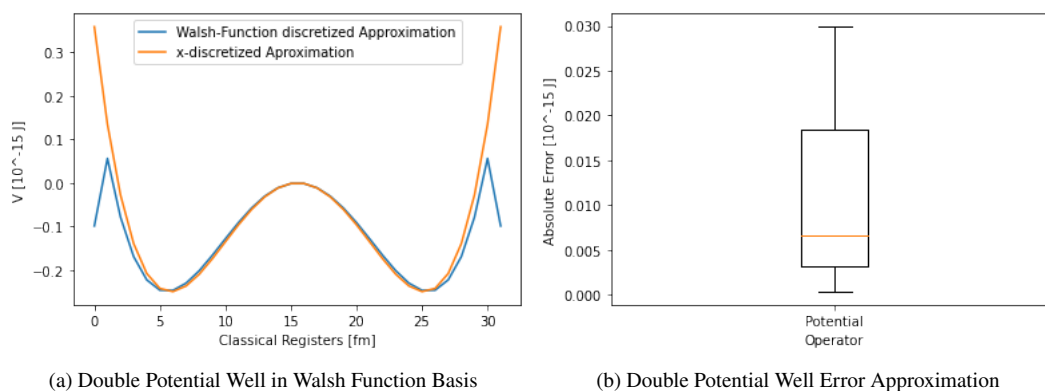


Figure 4.11: Double Potential Well Approximation

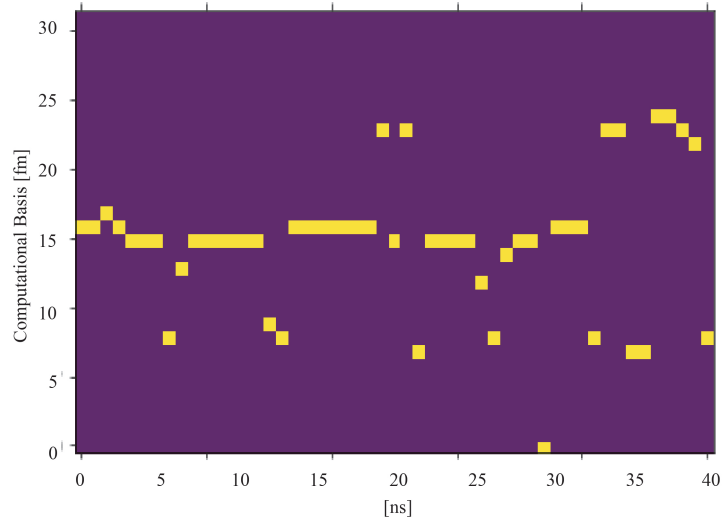
As in the Quantum Harmonic Oscillator, we will initialize the function using Dirac's delta function in the 4-qubit. The circuit implementation has the same structure as in Figure 4.4, and their potential, QFT, kinetic, and IQFT circuit implementations are attached in the Annex. We will take a $\Delta t = 0.1$ [ns], $t = 40$ [ns], and it is run into the state vector simulator as it is shown in Figure 4.12.

4.3.1 Discussion

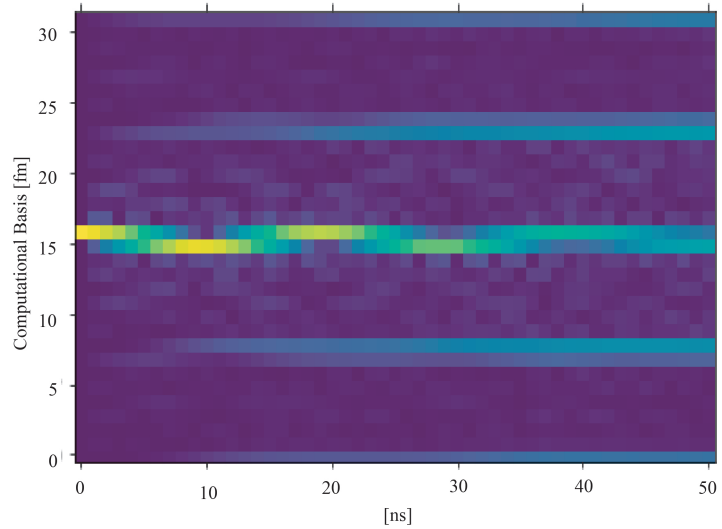
In Figure 4.12, we can see how a particle represented by a delta function evolve in a physical system where a double-well potential represents its potential part. At the beginning of its time steps, it can be shown how, after the third time step, the initial position is being changed between the computational state 14 (01110) and the state 15 (01111) as it is shown in the simulation of the Manchester Particle Physics Group 48. We can interpret that if it is not an exact energy eigenstate, it is a superposition of two nearly degenerate states, and as such, it will evolve slowly with time. After $t = 40$ [ns], we can see the initial state in its first excited state, having a more noticeable superposition of two degenerate states, separated in their position mapped on the fifth and 25th classical registers.

After that, we run our circuit into the ibmq-quito and ibmq-lima quantum computers for $\Delta t = 0.1$ [ns] and $t = 40$ [ns] with 1000 shots which were averaged and the final result is shown in Figure 4.13, we can notice how states' position evolves in time. Similarly to the Quantum Harmonic Oscillator, we compare the simulation executed with the state vector simulator and the ibmq-quito and ibmq-lima final measurements. This comparison is developed by using the classical fidelity measurement.

In Figure 4.13, we can see how the states' probabilities of the double-well potential evolve. During the three first columns, we can notice more explicitly how the behavior simulated in Ref. 48 is present. It shows two nearly



(a) Histogram Evolution Matrix



(b) Statevector Evolution Matrix

Figure 4.12: Double Potential Well Evolution in the Statevector Simulator with $\Delta t = 0.1$ [ns] and $t = 40$ [ns]

degenerate states of oscillation. At the same time, in the fourth column, the superposition of these two states is taking place along time slowly. This shows us how the bit-flips are produced as the system evolves. Finally, in the fourth column, we can see some quantum tunneling effect between the two potential wells, as expected in this system.

Analyzing the ibmq-lima and ibmq-quito final states' probabilities, we can notice that the initial state spreads along with the computational basis after the first time, similar to the QHO system. By analyzing the fidelity fluctuation along time, as shown in Figure 4.14, we notice that the behavior of both quantum machines is cyclical and has a parabolic behavior. The maximum fidelity obtained in ibmq-lima is 50.49 at $t = 21[ns]$, and in ibmq-quito is 49.09 at $t = 21[ns]$. Nonetheless, both devices' fidelities fluctuations have the same pattern. The processor's performance can be discussed as the ibmq-quito has a better one represented by a higher Quantum Volume factor. Thus it may suggest that it is necessary to let the system run for a more extensive t to develop a complete performance.

Regarding the computational performance of this system represented by a Double Well Potential, all of these circuits can be run in a linear time. Moreover, it means that this circuit implementation can be efficient compared with the classical ones. Finally, we can compare our simulation developed by using the UQS and other previous work as in Ref. 49 where the final probabilities show the same behavior as in Ref. 48.

4.4 Error Analysis

The obtained fidelity from the Quantum Harmonic Oscillator and the Double Well Potential are around the 50s percent. This is because of some hardware, software, and computational factors. Thus, let us analyze why this happens. At the very least, there will be a sampling error in going from the continuous $|x\rangle$ to the discrete $|x_k\rangle$ representation. Additionally, the Trotter error from splitting the propagator into non-commuting parts introduces further errors. Finally, by using (2.12), the total simulation error satisfies:

$$E(U, e^{-iHt}) \leq \alpha t \delta t + E_G \quad (4.3)$$

where $\alpha t \delta t$ is the first order Trotter error, and E_G denotes the gate error in evaluating the kinetic and potential propagators. We shall take into account that $E_G \leq \epsilon_v t + \epsilon_k t$ which belong to the approximation errors of the potential and kinetic part, correspondingly.

Fidelity Fluctuations								
Quantum Harmonic Oscillator			Double Well Potential					
Time Steps	ibmq-quito	ibmq-lima	Time Steps	ibmq-quito	ibmq-lima	Time Steps	ibmq-quito	ibmq-lima
1	30.82	32.71	1	30.82	36.47	21	49.09	50.49
2	35.66	34.72	2	33.99	36.26	22	44.30	43.73
3	38.21	39.06	3	35.69	39.54	23	40.49	46.02
4	40.82	40.92	4	33.10	34.24	24	43.19	43.75
5	45.31	43.41	5	37.74	33.99	25	41.85	40.48
6	45.37	42.95	6	32.48	36.51	26	36.80	41.31
7	46.73	44.36	7	30.54	30.95	27	36.36	35.68
8	47.82	41.81	8	33.68	35.02	28	34.89	36.30
9	50.82	48.57	9	31.70	27.17	29	36.60	35.01
10	52.82	49.98	10	23.23	24.07	30	31.98	33.75
11	52.61	47.21	11	23.96	25.17	31	30.64	31.96
12	52.00	52.83	12	32.76	35.59	32	37.75	41.72
13	48.75	45.48	13	35.13	30.66	33	40.16	38.16
14	47.08	44.12	14	35.14	36.25	34	40.90	42.20
15	44.16	40.90	15	39.86	39.33	35	42.19	42.01
16	35.52	36.35	16	46.28	48.76	36	44.07	47.07
17	38.64	30.79	17	43.64	43.25	37	44.82	45.51
18	37.65	35.51	18	40.23	43.99	38	42.49	46.29
19	38.91	37.47	19	39.46	43.45	39	40.55	45.10
20	46.19	42.63	20	43.25	42.46	40	47.87	47.48

Table 4.2: Experimental fidelity fluctuations obtained from each time running in ibmq-quito and ibmq-lima from the Quantum Harmonic Oscillator, and the Double Well Potential.



Figure 4.13: Double well potential evolution state's probabilities by using $\Delta t = 0.1$ [ns], $t = 40$ [ns], and 1000 shots executed on ibmq-quito and ibmq-lima

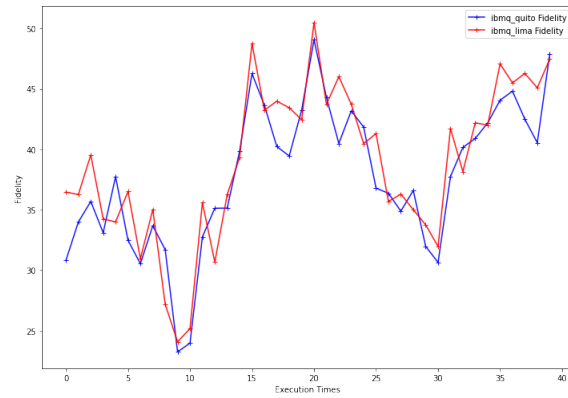


Figure 4.14: Double Well Potential Fidelity along Time Evolution

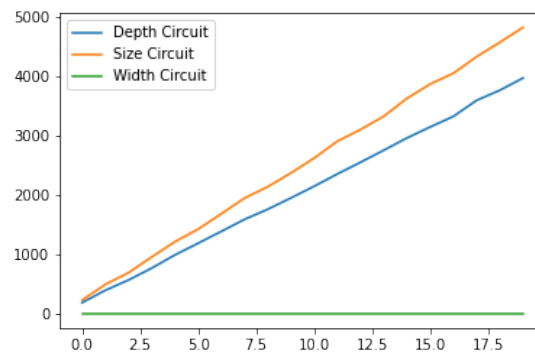


Figure 4.15: Double Well Potential Performance Indicators

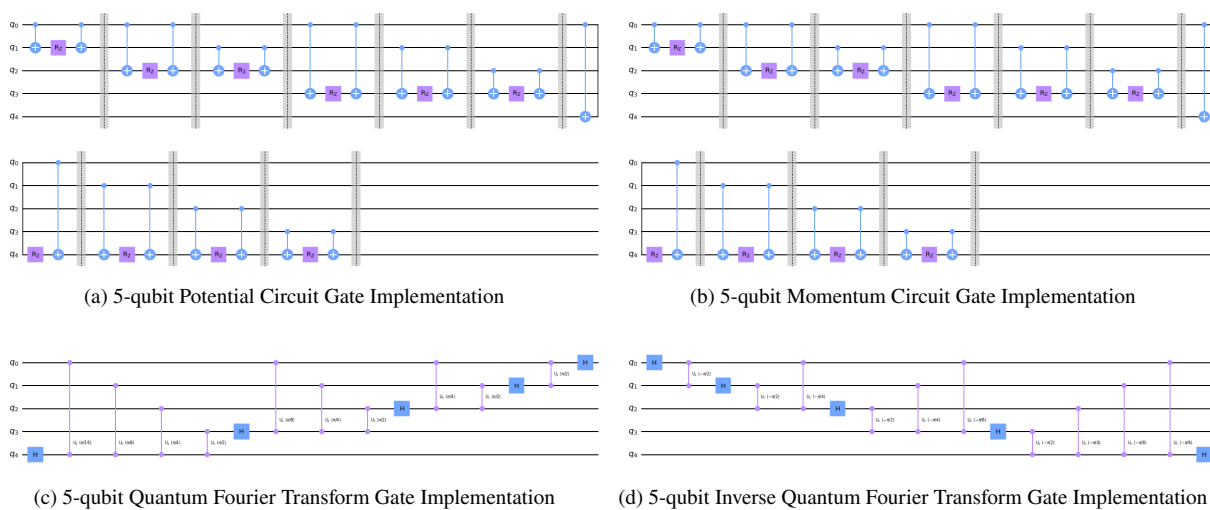


Figure 4.16: QHO Modular Circuit Implementation

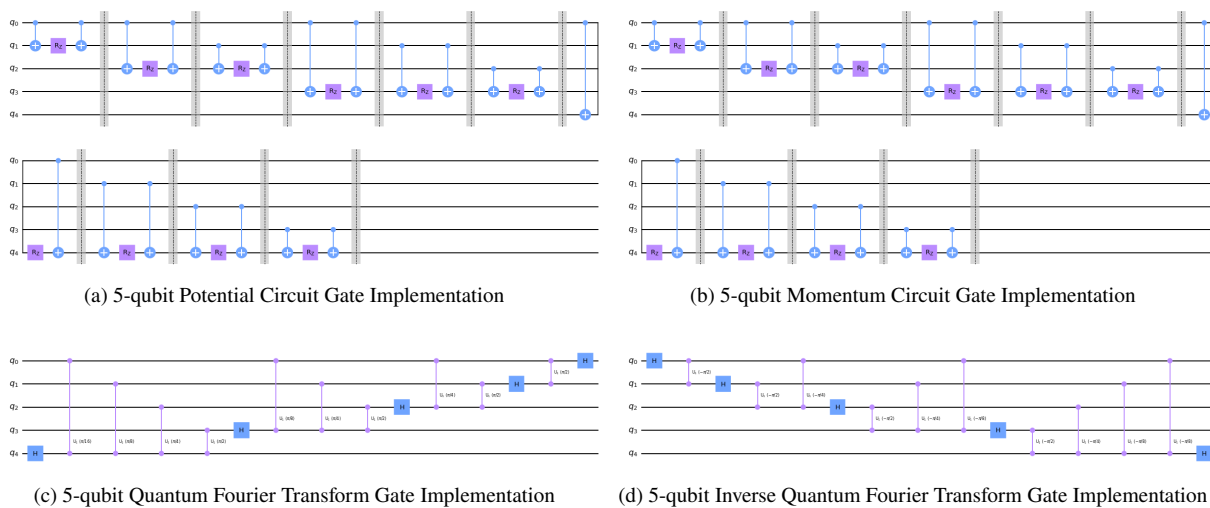
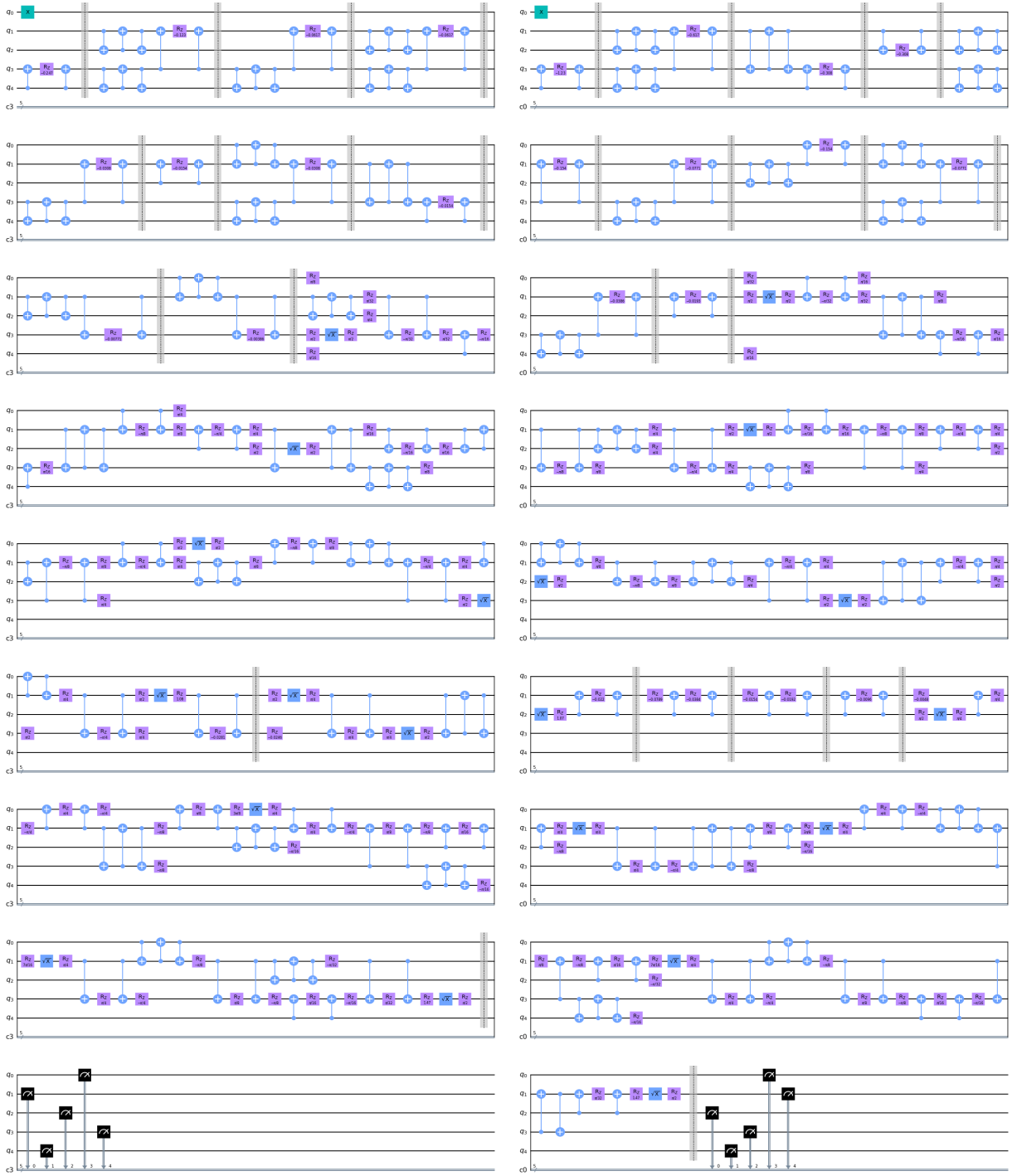


Figure 4.17: Double Well Potential Modular Circuit Implementation



(a) Quantum Harmonic Oscillator Transpilled Circuit.

(b) Double Well Potential Transpilled Circuit.

Figure 4.18: Transpilled Circuits in function of the ibmq-quantum computers basis gates. In ibmq-quito and ibmq-lima cases, the basis gates are the same: CX, ID, RZ, SX, X.

Chapter 5

Conclusions & Outlook

In this work, we have developed and implemented a Universal Quantum Simulator by using the gate circuit building methodology. The main feature of this simulator is that the gate implementation is automatic, as it is only necessary to place the Hamiltonian representation that describes the physical system. After that, we discrete and expand the Hamiltonian constituents on Walsh functions and proceed to use the group $Z^{\otimes n}$, Walsh operators, to the gate implementation. All this process is done by using the qiskit platform.

In order to test the feasibility of the UQS, we use two 1D well-known systems: Quantum Harmonic Simulator and the Double Well Potential as the extensive research done upon these two physical systems and the possibility that it gives us to test the feasibility of the Universal Quantum Simulator. However, we can simulate in more than 1D by increasing the number of simulated variables. At first, select the number of qubits where their absolute error distribution is the lowest possible. Thus, we work with a five qubits system. As the initial state, we use a Dirac function represented by a X gate representing a bit-flip. Our initial state is placed in the 4 qubit in both cases. After that, we set $\Delta t = 0.1 [ns]$ as the time steps; we let the systems evolve along with time and position in the state vector simulator and the ibmq-quito and ibmq-lima quantum processors.

Using the state vector simulator, the (QHO) showed how the initial state was evolving and the presence of the ground, first, and a linear combination of both energy states associated with each state. These energy states were shown by identifying a bit-flip along time. In the same way, the second simulated physical system (Double Well Potential) showed an initial ground state, which evolved from previous classical models. Over time, there appeared the initial state in its first energy state. Additionally, we captured the tunneling happening in each potential well in this system, as predicted initially.

After that, we run the circuits of each system in the ibmq-lima and ibmq-quito processors. In the QHO, the maximum fidelities obtained were 52.83 at $t = 12 [ns]$ and 52.60 $[ns]$ at $t = 9 [ns]$, correspondingly. after some time evolution. In the DWP, the maximum fidelities obtained were 50 at $t = 23 [ns]$ and 49 at $t = 23 [ns]$. In both cases, the running time grows linearly as its depth circuit. The final fidelities, calculated using the classical definition of fidelity, were affected by errors from hardware, software, and computational factors. Thus, in the hardware field, the performance of each quantum processor will be affected by their technical characteristics summed up into the

Quantum Volume parameters. The error can be associated with the IBM-quantum-experience platform calibrations of each qubit, which are done periodically. Furthermore, regarding the contribution of the computational error, we can notice that the x -discretization process carries an error, the Trotterization process, and the gate error in evaluating each kinetic and potential operator.

Finally, as an option to increase the fidelity, it would be interesting to use a Variational Quantum Eigensolver as a method to capture the state evolution in each time evolution; as a result, we have an increasing the fidelity, decreasing the circuit depth size, and have an optimized size circuit and consequently higher fidelities. Furthermore, error correction techniques can also be used to encode information as a tool to keep a high fidelity. Moreover, it will be interesting to explore the UQS functionality with more quantum physical systems in different dimensions.

Appendix A

Quantum Fourier Transform Gate Representation

$$\begin{aligned} |x\rangle &\rightarrow \frac{1}{2^{n/2}} \sum_{k=0}^{2^n-1} e^{2\pi i x k / 2^n} |k\rangle \\ &= \frac{1}{2^{n/2}} \sum_{k_1=0}^0 \dots \sum_{k_n=0}^0 e^{2\pi i x (\sum_{l=1}^n k_l 2^{l-1})} |k_1 \dots k_n\rangle \\ &= \frac{1}{2^{n/2}} \sum_{k_1=0}^0 \dots \sum_{k_n=0}^0 \otimes_{l=1}^n e^{2\pi i x k_l 2^{l-1}} |k_l\rangle \\ &= \frac{1}{2^{n/2}} \otimes_{l=1}^n \left[\sum_{k_l=0}^1 e^{2\pi i x k_l 2^{l-1}} |k_l\rangle \right] \\ &= \frac{1}{2^{n/2}} \otimes_{l=1}^n \left[|0\rangle + e^{2\pi i x 2^{l-1}} |1\rangle \right] \\ &= QFT_N |x\rangle = \frac{1}{\sqrt{N}} (|0\rangle + e^{2\pi i x / 2} |1\rangle) \otimes (|0\rangle + e^{2\pi i x / 2^2} |1\rangle) \otimes \dots \otimes (|0\rangle + e^{2\pi i x / 2^n} |1\rangle) \end{aligned}$$

Appendix B

Simulation of the Quantum Harmonic Oscillator on Matlab

Let us consider an Harmonic Oscillator Hamiltonian given by:

$$\hat{H} = \frac{\hat{P}^2}{2m} + \frac{1}{2}m\omega^2 Q^2 = \left(\hat{a}^\dagger \hat{a} + \frac{1}{2}\right)\hbar\omega \quad (\text{B.1})$$

formed by the kinetic and potential energy. Its time dependant solution is given by:

$$\hat{H} |n\rangle = E_n |n\rangle \quad (\text{B.2})$$

$$E_n = \hbar\omega \left(n + \frac{1}{2}\right) \quad (\text{B.3})$$

where the annihilation and creation operators are given by:

$$\hat{a} |n\rangle = \begin{cases} 0 & n = 0 \\ \sqrt{n} |n-1\rangle & \text{otherwise} \end{cases}$$
$$\hat{a}^\dagger |n\rangle = \sqrt{n+1} |n+1\rangle$$

We can write the general state as a superposition of eigenstates as in B.2. Thus, we can generate a matrix representation of diverse eigenstates by using the annihilation and creation matrices:

$$\hat{a} \leftrightarrow \tilde{a} = \begin{bmatrix} 0 & 1 & 0 & 0 & \dots \\ 0 & 0 & \sqrt{2} & 0 & \dots \\ 0 & 0 & 0 & \sqrt{3} & \dots \\ 0 & 0 & 0 & 0 & \dots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \quad (\text{B.4})$$

$$\hat{a}^\dagger \leftrightarrow \tilde{a}^\dagger = \begin{bmatrix} 0 & 0 & 0 & 0 & \dots \\ 1 & 0 & 0 & 0 & \dots \\ 0 & \sqrt{2} & 0 & 0 & \dots \\ 0 & 0 & \sqrt{3} & 0 & \dots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \quad (\text{B.5})$$

Finally, our Hamiltonian is represented as:

$$\widehat{H} \leftrightarrow \widetilde{H} = \hbar\omega\left(\hat{a}^\dagger\hat{a} + \frac{1}{2}\mathbb{I}\right) = \hbar\omega \begin{bmatrix} 1 & 0 & 0 & 0 & \dots \\ 0 & 3 & 0 & 0 & \dots \\ 0 & 0 & 5 & 0 & \dots \\ 0 & 0 & 0 & 7 & \dots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \quad (\text{B.6})$$

We have plot this model in *Matlab* and the different energy states of our function are plotted in Figure 4.7.

Appendix C

Errors of Walsh Function Approximation for more than 6 qubits

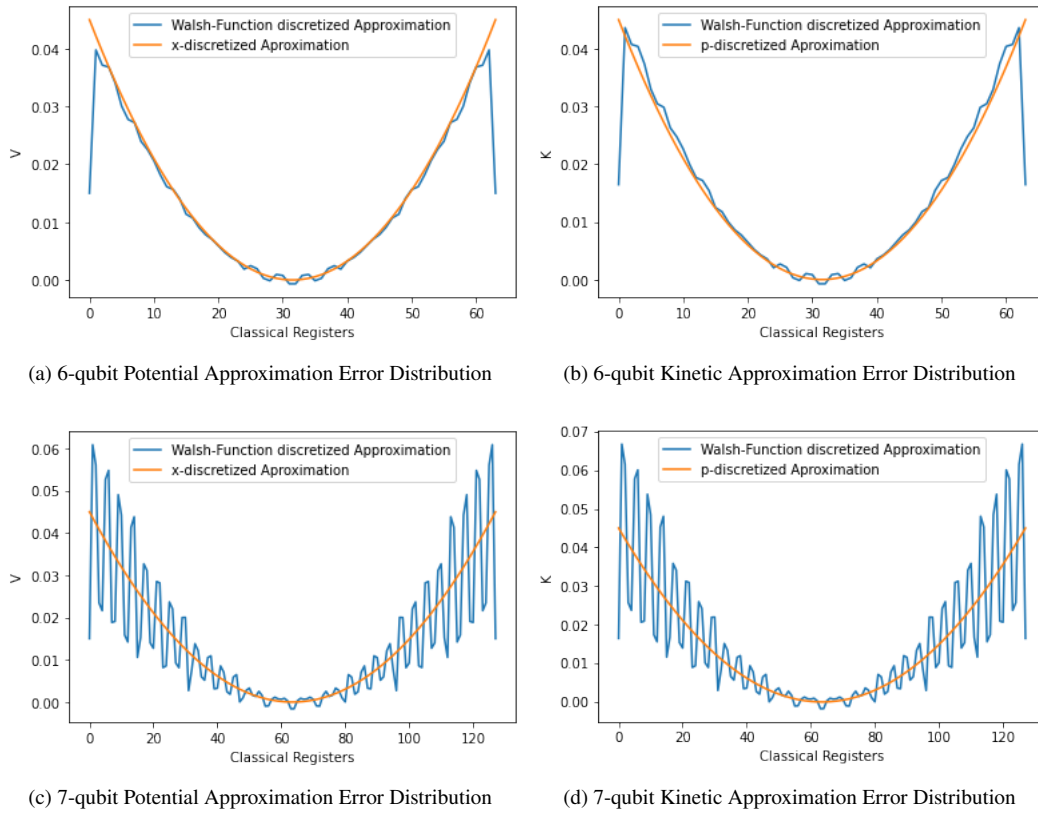


Figure C.1: Operator Approximation Error Distributions for 6 and 7 qubits.

Bibliography

- [1] IBM, IBM Quantum. <https://quantum-computing.ibm.com/>.
- [2] Pérez-Delgado, C. A.; Kok, P. Quantum computers: Definition and implementations. *Physical Review A* **2011**, *83*, 012303.
- [3] Nielsen, M. A.; Chuang, I. Quantum computation and quantum information. 2002.
- [4] Tillmann, M.; Dakić, B.; Heilmann, R.; Nolte, S.; Szameit, A.; Walther, P. Experimental boson sampling. *Nature photonics* **2013**, *7*, 540–544.
- [5] Xu, N.; Zhu, J.; Lu, D.; Zhou, X.; Peng, X.; Du, J. Quantum factorization of 143 on a dipolar-coupling nuclear magnetic resonance system. *Physical review letters* **2012**, *108*, 130501.
- [6] Zhang, Y.-S.; Li, C.-F.; Guo, G.-C. Quantum key distribution via quantum encryption. *Physical Review A* **2001**, *64*, 024302.
- [7] Rodenburg, B.; Pappas, S. P. *Blockchain and quantum computing*; 2017.
- [8] Gao, Y.-L.; Chen, X.-B.; Chen, Y.-L.; Sun, Y.; Niu, X.-X.; Yang, Y.-X. A secure cryptocurrency scheme based on post-quantum blockchain. *IEEE Access* **2018**, *6*, 27205–27213.
- [9] Egger, D. J.; Gambella, C.; Marecek, J.; McFaddin, S.; Mevissen, M.; Raymond, R.; Simonetto, A.; Woerner, S.; Yndurain, E. Quantum computing for finance: State-of-the-art and future prospects. *IEEE Transactions on Quantum Engineering* **2020**, *1*, 1–24.
- [10] Herman, D.; Googin, C.; Liu, X.; Galda, A.; Safro, I.; Sun, Y.; Pistoia, M.; Alexeev, Y. A Survey of Quantum Computing for Finance. *arXiv preprint arXiv:2201.02773* **2022**,
- [11] Bova, F.; Goldfarb, A.; Melko, R. G. Commercial applications of quantum computing. *EPJ quantum technology* **2021**, *8*, 2.
- [12] Ajagekar, A.; You, F. Quantum computing for energy systems optimization: Challenges and opportunities. *Energy* **2019**, *179*, 76–89.

- [13] Wang, L.; Tang, F.; Wu, H. Hybrid genetic algorithm based on quantum computing for numerical optimization and parameter estimation. *Applied Mathematics and Computation* **2005**, *171*, 1141–1156.
- [14] Marzec, M. Portfolio optimization: Applications in quantum computing. *Handbook of high-frequency trading and modeling in finance* **2016**, 73–106.
- [15] ANIS, M. S.; Abraham, H.; AduOffei,; Čepulkovskis, M. Qiskit: An Open-source Framework for Quantum Computing. 2021.
- [16] Skibba, R. Einstein, Bohr and the war over quantum theory. *Nature* **2018**, *555*, 582–585.
- [17] Clauser, J. F.; Shimony, A. Bell's theorem. Experimental tests and implications. *Reports on Progress in Physics* **1978**, *41*, 1881.
- [18] Ashkin, A. Atomic-beam deflection by resonance-radiation pressure. *Physical Review Letters* **1970**, *25*, 1321.
- [19] Hou, S.-Y.; Sheng, Y.-B.; Feng, G.-R.; Long, G.-L. Experimental optimal single qubit purification in an NMR quantum information processor. *Scientific reports* **2014**, *4*, 1–7.
- [20] Oliveira, I.; Sarthour Jr, R.; Bonagamba, T.; Azevedo, E.; Freitas, J. C. *NMR quantum information processing*; Elsevier, 2011.
- [21] Treutlein, P.; Steinmetz, T.; Colombe, Y.; Lev, B.; Hommelhoff, P.; Reichel, J.; Greiner, M.; Mandel, O.; Widera, A.; Rom, T. Quantum information processing in optical lattices and magnetic microtraps. *Fortschritte der Physik: Progress of Physics* **2006**, *54*, 702–718.
- [22] Lewenstein, M.; Sanpera, A.; Ahufinger, V. *Ultracold Atoms in Optical Lattices: Simulating quantum many-body systems*; OUP Oxford, 2012.
- [23] Almeida, G. M.; Ciccarello, F.; Apollaro, T. J.; Souza, A. M. Quantum-state transfer in staggered coupled-cavity arrays. *Physical Review A* **2016**, *93*, 032310.
- [24] Hartmann, M. J.; Brandao, F. G.; Plenio, M. B. Quantum many-body phenomena in coupled cavity arrays. *Laser & Photonics Reviews* **2008**, *2*, 527–556.
- [25] Mei, F.; Guo, Q.; Yu, Y.-F.; Xiao, L.; Zhu, S.-L.; Jia, S. Digital simulation of topological matter on programmable quantum processors. *Physical Review Letters* **2020**, *125*, 160503.
- [26] Boothby, K.; Bunyk, P.; Raymond, J.; Roy, A. Next-generation topology of d-wave quantum processors. *arXiv preprint arXiv:2003.00133* **2020**,
- [27] Hendrickx, N. W.; Lawrie, W. I.; Russ, M.; van Riggelen, F.; de Snoo, S. L.; Schouten, R. N.; Sammak, A.; Scappucci, G.; Veldhorst, M. A four-qubit germanium quantum processor. *Nature* **2021**, *591*, 580–585.

- [28] Watson, T.; Philips, S.; Kawakami, E.; Ward, D.; Scarlino, P.; Veldhorst, M.; Savage, D.; Lagally, M.; Friesen, M.; Coppersmith, S. A programmable two-qubit quantum processor in silicon. *Nature* **2018**, *555*, 633–637.
- [29] Paesani, S.; Laing, A. *Silicon Photonics IV*; Springer, 2021; pp 449–489.
- [30] Kyaw, T. H.; Menke, T.; Sim, S.; Anand, A.; Sawaya, N. P.; Oliver, W. D.; Guerreschi, G. G.; Aspuru-Guzik, A. Quantum computer-aided design: digital quantum simulation of quantum processors. *Physical Review Applied* **2021**, *16*, 044042.
- [31] Bell, J. S. On the einstein podolsky rosen paradox. *Physics Physique Fizika* **1964**, *1*, 195.
- [32] Patrick, K. Which is greater? The number of atoms in the universe or the number of chess moves? *National Museum Liverpool*
- [33] Sparkes, M. A new quantum leader? 2021.
- [34] Deutsch, D. Quantum theory, the Church–Turing principle and the universal quantum computer. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences* **1985**, *400*, 97–117.
- [35] Hales, L.; Hallgren, S. An improved quantum Fourier transform algorithm and applications. 2000.
- [36] Ambainis, A. Quantum search algorithms. *ACM SIGACT News* **2004**, *35*, 22–35.
- [37] Kitaev, A. Y.; Shen, A.; Vyalıy, M. N.; Vyalıy, M. N. *Classical and quantum computation*; American Mathematical Soc., 2002.
- [38] Griffiths, D. J. *Introduction to quantum mechanics*; Pearson International Edition (Pearson Prentice Hall, Upper Saddle River, 2005), 1962.
- [39] Weideman, J.; Herbst, B. Split-step methods for the solution of the nonlinear Schrödinger equation. *SIAM Journal on Numerical Analysis* **1986**, *23*, 485–507.
- [40] Lloyd, S. *Quantum information with continuous variables*; Springer, 2003; pp 37–45.
- [41] Suzuki, M. Improved Trotter-like formula. *Physics Letters A* **1993**, *180*, 232–234.
- [42] Zalka, C. Simulating quantum systems on a quantum computer. *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences* **1998**, *454*, 313–322.
- [43] Welch, J.; Greenbaum, D.; Mostame, S.; Aspuru-Guzik, A. Efficient quantum circuits for diagonal unitaries without ancillas. *New Journal of Physics* **2014**, *16*, 033040.
- [44] Golubov, B.; Efimov, A.; Skvortsov, V. *Walsh series and transforms: theory and applications*; Springer Science & Business Media, 2012; Vol. 64.

-
- [45] Moll, N.; Barkoutsos, P.; Bishop, L. S.; Chow, J. M.; Cross, A.; Egger, D. J.; Filipp, S.; Fuhrer, A.; Gambetta, J. M.; Ganzhorn, M. Quantum optimization using variational algorithms on near-term quantum devices. *Quantum Science and Technology* **2018**, 3, 030503.
- [46] Jain, V. K.; Behera, B. K.; Panigrahi, P. K. Quantum Simulation of Discretized Harmonic Oscillator on IBMQuantum Computer. **2019**,
- [47] Jelic, V.; Marsiglio, F. The double-well potential in quantum mechanics: a simple, numerically exact formulation. *European Journal of Physics* **2012**, 33, 1651.
- [48] HEP-Group, An example to illustrate how indistinguishable particles can behave as if they are distinguishable. 2016; <https://www.hep.manchester.ac.uk/u/forshaw/BoseFermi/Double2Well.html>.
- [49] Hegade, N. N.; Behera, B. K.; Panigrahi, P. K. Experimental demonstration of quantum tunneling in IBM quantum computer. *arXiv preprint arXiv:1712.07326* **2017**,