Quantum Computing Concepts with Qubit Manipulation

Carin Samer , Aya Ahmed

June 2024

Abstract

From simple wooden devices to digital electronic computers to supercomputers, physics and computation have always been intertwined. This is due to the fact of using physical systems in encoding information and processing them. Quantum computing is a research area that extends the set of physical laws classical computers operate on by accessing the quantum aspects of a physical world, opening up new ways for processing information.

The ability to simulate and compute is enhanced by increasing the number of transistors, as their number directly affects processing and accuracy capability. According to Moore's Law, the maximum number of transistors in an integrated circuit doubles approximately every two years. As soon as it doubles, computer performance increases. However, there's a theoretical limit to the miniaturization of computing components. We can't make a transistor smaller than an atom! This is where quantum computing comes into play.

When Richard Feynman first proposed the notion of a quantum computer in 1982, his focus was on simulating quantum mechanical systems. To the best of our understanding, nature is not classical — it's quantum. And so, quantum computers may prove to be valuable tools for understanding it.

1 Introduction

According to Moore's Law, the maximum number of transistors in an integrated circuit doubles approximately every two years. As soon as it doubles, computer performance increases. However, there's a theoretical limit to the miniaturization of computing components. We can't make a transistor smaller than an atom! This is where quantum computing comes into play. The need for a quantum computer stems from the inability of classical computers to solve "complex problems" as modeling the atomic behavior which is essential for many purposes as drug manufacturing. When Richard Feynman first proposed the notion of a quantum computer in 1982, his focus was on simulating quantum mechanical systems saying that: "Nature does not behave in a classical way, so it is impossible to simulate it with classical computer." A Quantum Computer is a special kind of computers that utilizes the quantum mechanical principles to perform computations depending on qubits.

2 Methodology of Working of a Quantum Computer

2.1 Quantum Bits

It is the quantum version of the classic binary bit with a two-state (or two-level) quantum-mechanical system to process information with more powerful computational power thanks to the quantum properties as entanglement and superposition. In classical computers, whether electrons move or not through a conventional transistor determines the bit's value, making it either 1 or 0. But in a quantum computer rather than simply switching electron flow on or off to encode information, it requires control over pure quantum properties like electron spin to be the two-level encoding system.

Qubits are typically represented by quantum states of physical systems as :

Physical System	**Encoding State(s)**
Superconductors QC (Electron)	Electron Spin
Linear Optical QC (Photons)	Path of photons
Neutral atoms in a lattice QC	Hyperfine Energy level or excited state

In order to get the most out of the great quantum properties , you need to have a fine grip over a material's electrons or other quantum particles . To create a qubit, scientists have to find a spot in a material where they can access and control these quantum properties. Once they access them, they can then use light or magnetic fields to create superposition, entanglement, and other properties. Whether they use electron spin or another approach, all qubits face major challenges before we can scale them up. Two of the biggest ones are coherence time and error correction.

2.2 Unitary and Non-unitary

They are fundamental operations in quantum computing like logic gates in classical computers, they manipulate qubits by altering their quantum states. To mathematically determine the effect of a gate on a qubit state, we use the formalism of quantum mechanics, where quantum gates are represented by unitary matrices and qubit states are represented by vectors (state vectors) in a Hilbert space. Quantum gates split into two categories:

Unitary Gates	**Non-Unitary Gates**
A unitary gate is a quantum gate represented by a unitary matrix Unitary gates preserve the norm (or total probability) of quantum states. The vectors representing quantum states have the same inner product after applying the gate.	An non- unitary gate is a gate that does not satisfy the unitary condition. Non-unitary operations can include measurement, decoherence, and noise processes, which are inherently irreversible. Non unitary operations cause loss of information and non preservation of the norm of the quantum state.
E.g. Pauli-X , Pauli-Y, Pauli-Z, Hadamard Gate and CNOT gate.	E.g. Measurement gate, Amplitude Damping Gate, Phase Damping Gate and Depolarization Channel.

2.2.1 Unitary Gates

Single Qubit Gates	Multi-Qubit Gates
Work on each qubit independently	Work on multiple qubits simultaneously
E.g. Pauli-X , Pauli Y and Pauli-Z	E.g. CNOT gate

2.3 Qubit Manipulation

Control is critical for qubit initialization , manipulation, readout and low latency feedback for error correction. A key requirement for optimal qubit processor performance is the ability of controlling qubits robustly and cost effectively.

Qubit are controlled using high-frequency (either microwave or laser) signals . Most architectures approach the challenge of parallelising qubit control by supplying independent high-frequency signals to each qubit.

When controlling a qubit we assume that the property that we work on,like the spin of an electron- is a vector in the Hilbert space and we manipulate this property by performing unitary operations (quantum gates , which we will tackle later) on the vector in the Hilbert space. Interaction with a resonant microwave signal causes rotation of the state vector in the Hilbert space thus the qubit state is controlled. To determine the effect of a gate on a qubit state, follow these steps:

1. Represent the initial qubit state as a vector
2. Represent the quantum gates as a unitary matrix
3. Multiply the matrix by the vector to find the new state.

This linear algebraic approach allows for precise calculation of the qubit states after the application of quantum gates.

3 Mathematical Underpinnings of Qubits

3.1 How qubit states are represented as vectors in a 2D complex vector space

Quantum computing power arises, in part, because the dimension of the vector space of quantum state vectors grows exponentially with the number of qubits. This means that while a single qubit can be trivially modeled, simulating a fifty-qubit quantum computation would arguably push the limits of existing supercomputers.

In quantum computing , the state vector of a qubit represents its state in a complex vector space . it's typically denoted as $|\Psi\rangle$ and can be expressed as a linear combination of the basis states $|0\rangle$ and $|1\rangle$:

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

The states $|0\rangle$and $|1\rangle$ are usually represented as

$$|0\rangle = \begin{bmatrix}1\\0\end{bmatrix}, |1\rangle = \begin{bmatrix}0\\1\end{bmatrix}$$

Here, and are complex numbers that represent the probability amplitudes for the qubit being in state $|0\rangle$ or $|1\rangle$, respectively. The probabilities of measuring the qubit in either state are given by the squares of the magnitudes of these amplitudes:

$$P(0) = |\alpha|^2, P(1) = |\beta|^2$$

The state vector must also satisfy the normalization condition which is know as "Born's rule":

$$|\alpha|^2 + |\beta|^2 = 1$$

Experimentally, a qubit's state can be changed through some physical action such as applying an electromagnetic laser or passing it through an optical device. Changing a qubit's state through a physical action mathematically corresponds to multiplying the qubit vector $|\Psi\rangle$ by some **unitary matrix U (which corresponds to applying quantum gates).** After this operation , the state $|\Psi\rangle$ becomes $U\,|\Psi\rangle$.

Unitary is a mathematical term which expresses that U can only act on the qubit in such a way that the total probability $|\alpha|^2 + |\beta|^2$ does not change. This is very important because, in all mathematical constructions of quantum mechanics, **one fundamental assumption is that each (matrix) operator must be unitary.** This ensures that after changing any state through an action, the total probability to observe all possible states will still add up to 100%. If this did not happen, then we could not interpret the results of quantum mechanics to be probabilistic, and the results would disagree with the many experiments that have been performed to date. The physical action of interacting with the state corresponds mathematically to applying a unitary operator.

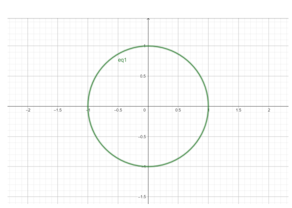
4 Qubit Representation

4.1 The Bloch Sphere

The Bloch sphere is a geometric representation of the quantum state of a two-level quantum system, such as a qubit. It provides a clear visualization of the quantum states and their properties. In the context of the Bloch sphere, the zero and one states are represented by specific points on the sphere's surface. These points are known as antipodal states due to their positioning on opposite sides of the sphere.

The Bloch Sphere is a generalization of the representation of a complex number z with $|z|^2 = 1$ as a point on the unit circle in the complex plane. If $z = x + iy$, where x and y are real, then:

$$|z|^2 = z * z = (x - iy)(x + iy) = x^2 + y^2$$



and $x^2 + y^2$ is the equation of a circle of radius one, centered on the origin.

4.2 Rotations on the Bloch Sphere

The Pauli X, Y and Z matrices are so-called because when they are exponentiated, they give rise to the rotation operators, which rotate the Bloch vector $(\sin\theta\cos\phi, \sin\theta\sin\phi, \cos\theta)$ about the $\vec{x}, \vec{y} and \vec{z}$ axes:

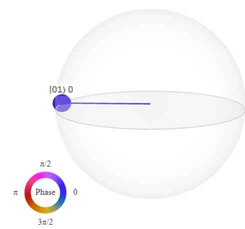
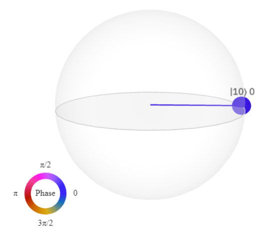
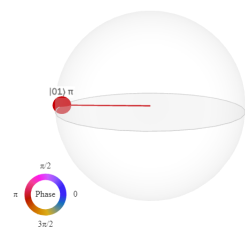
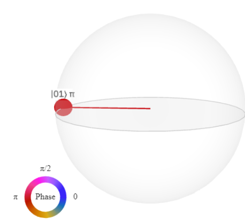
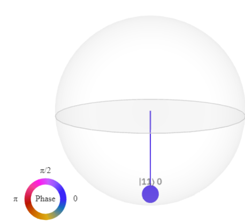
$$R(\theta) \equiv e^{-i\theta X/2}$$

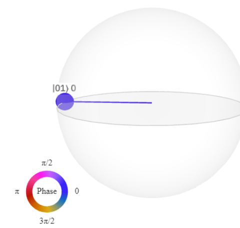
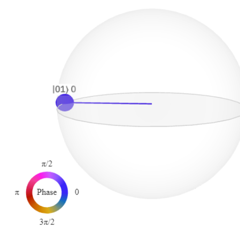
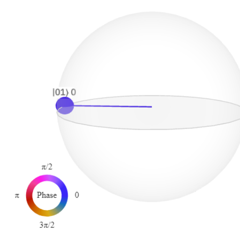
$$R(\theta) \equiv e^{-i\theta Y/2}$$

$$R(\theta) \equiv e^{-i\theta Z/2}$$

5 Gate Operations

This table illustrates the application of some unitary gates on the qubit states

Pauli-X Gate The toggling switch Flips the state of the qubit	$X\|0\rangle = \begin{bmatrix}0 & 1\\1 & 0\end{bmatrix}\begin{bmatrix}1\\0\end{bmatrix} = \begin{bmatrix}0\\1\end{bmatrix} = \|1\rangle$ $X\|1\rangle = \begin{bmatrix}0 & 1\\1 & 0\end{bmatrix}\begin{bmatrix}0\\1\end{bmatrix} = \begin{bmatrix}1\\0\end{bmatrix} = \|0\rangle$	
Pauli-Y Gate Rotates the qubit's phase by 180 degrees around the Y-axis.	$Y\|0\rangle = \begin{bmatrix}0 & -i\\i & 0\end{bmatrix}\begin{bmatrix}1\\0\end{bmatrix} = \begin{bmatrix}0\\i\end{bmatrix}$ $Y\|1\rangle = \begin{bmatrix}0 & -i\\i & 0\end{bmatrix}\begin{bmatrix}0\\1\end{bmatrix} = \begin{bmatrix}-i\\0\end{bmatrix}$	
Pauli-Z Gate Unveiling the Quantum Phase Adds a pi to the phase if the qubit is in the 1 state.	$Z\|0\rangle = \begin{bmatrix}1 & 0\\0 & -1\end{bmatrix}\begin{bmatrix}1\\0\end{bmatrix} = \begin{bmatrix}1\\0\end{bmatrix}$ $Z\|1\rangle = \begin{bmatrix}1 & 0\\0 & -1\end{bmatrix}\begin{bmatrix}0\\1\end{bmatrix} = \begin{bmatrix}0\\-1\end{bmatrix}$	
Hadamard Gate It is a Clifford gate that rotates the qubit's state 90 degrees around the X+Z axis. It also creates superposition.	$H\|0\rangle = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}\begin{bmatrix}1\\0\end{bmatrix} = \frac{\|0\rangle + \|1\rangle}{\sqrt{2}}$ $H\|1\rangle = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}\begin{bmatrix}0\\1\end{bmatrix} = \frac{\|0\rangle - \|1\rangle}{\sqrt{2}}$	
CNOT(CX) Gate A double qubit gate ; control qubit and target qubit. It XORs the controlling qubit and applies the new state on the target qubit.		

Rx-Gate It is a standard rotation gate that rotates the qubit's state 60 degrees around the X-axis	$R_X(\theta) = \begin{bmatrix} \cos\theta/2 & -i\sin\theta/2 \\ -i\sin\theta/2 & \cos\theta/2 \end{bmatrix}$	
Ry-Gate It is a standard rotation gate that rotates the qubit's state 90 degrees around the Y-axis	$R_Y(\theta) = \begin{bmatrix} \cos\theta/2 & -\sin\theta/2 \\ \sin\theta/2 & \cos\theta/2 \end{bmatrix}$	
Rz-Gate It is a standard rotation gate that rotates the qubit's state 180 degrees around the Z-axis	$R_Z(\theta) = \begin{bmatrix} \exp -i\theta/2 & 0 \\ 0 & \exp i\theta/2 \end{bmatrix}$	

From the previous table , we conclude the behavior of each quantum gate and the mathematical relation behnid it , and also the Bloch sphere representation of the results.

6 Applying Gates on Multiple Qubits

As we clarified in the previous section the how a unitary matrix (which is professionally known as quantum gates) can affect the state of a qubit, let's explore the math behind this phenomenon. Think of a quantum gate as an "effect" added to a qubit or a group of them.

To see this, consider measuring the first qubit of the following state, which is formed by applying the Hadamard transform H on two qubits initially set to the "0" state:

$$H^{\otimes 2}\left(\begin{bmatrix}1\\0\end{bmatrix} \otimes \begin{bmatrix}1\\0\end{bmatrix}\right) = \frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}\begin{bmatrix}1\\0\\0\\0\end{bmatrix} = \frac{1}{2}\begin{bmatrix}1\\1\\1\\1\end{bmatrix}$$

[1]

Both outcomes have a 50% probability of occurring (according to Born's Rule) . That can be intuited from the fact that the quantum state before measurement does not change if 0 is swapped with 1 on the first qubit.

Here's also another example on applying a Pauli-X gate on a set of 3 qubits initially set to the "1" state :

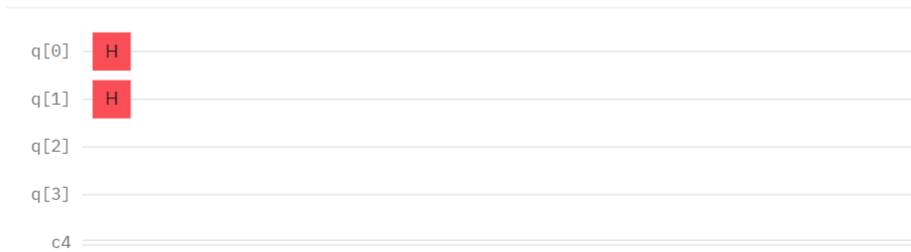
$$(X \otimes X \otimes X)\begin{bmatrix}0\\0\\0\\0\\1\\0\\0\end{bmatrix} = \begin{bmatrix}0 & 1\\1 & 0\end{bmatrix} \otimes \begin{bmatrix}0 & 1\\1 & 0\end{bmatrix} \otimes \begin{bmatrix}0 & 1\\1 & 0\end{bmatrix}\begin{bmatrix}0\\0\\0\\0\\1\\0\\0\end{bmatrix} = \begin{bmatrix}1\\0\\0\\0\\0\\0\\0\end{bmatrix}$$

[1] $\otimes$:The tensor product of two vector spaces captures the properties of all bilinear maps in the sense that a bilinear map from VxZ into another vector space Z factors uniquely through a linear map $V \otimes W \rightarrow Z$

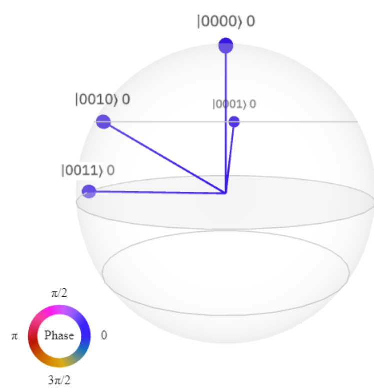
6.1 Analyzing the Composite Effects of Sequential Gate Operations on Qubit States

In the previous sections, positions of a qubit state on a Bloch sphere were clarified clearly. Now , let's take a look at the perspectives of applying a series of gates on multiple qubits. To explore the essence of quantum logic gates, let's explore their properties through the next experiment.

Initially , assume having 4 qubits and apply the Hadamard gate to the first a

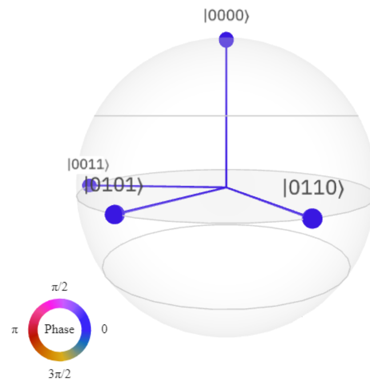


Results: Creating a superposition state on the first 2 qubits , obtaining 4 resulting states [0.5+0j, 0.5+0j, 0.5+0j, 0.5+0j]. Here is the Bloch Sphere that represents the results.



Then , add a controlled X gate (CNOT) gate to (q[2],q[0]) and (q[2],q[1])

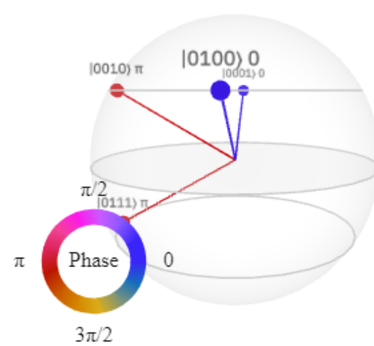
Results: Adding a CNOT gate XORs the state of the controlling qubit and applies it to the target qubit , resulting in $[0.5+0j, 0+0j, 0+0j, 0.5+0j, 0+0j, 0.5+0j, 0.5+0j, 0+0j, 0+0j, 0+0j, 0+0j, 0+0j, 0+0j, 0+0j, 0+0j, 0+0j]$



After that apply an X gate on q[0] and a Z-gate on q[1]

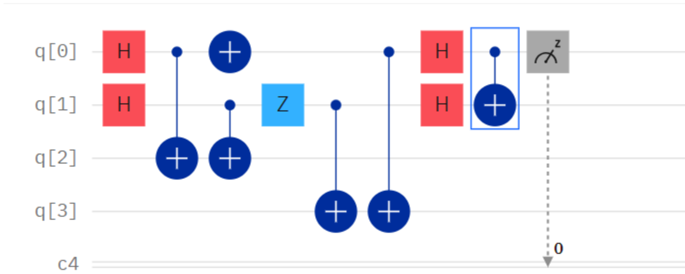
Results:

- The X-gate applied on q[0] flips its state to "1" and also affects q[2] as they're connected through a CNOT gate.
- The Z-gate applied to q[1] has no effect as the qubit is set to "0" after applying Hadamard gate in the previous step.

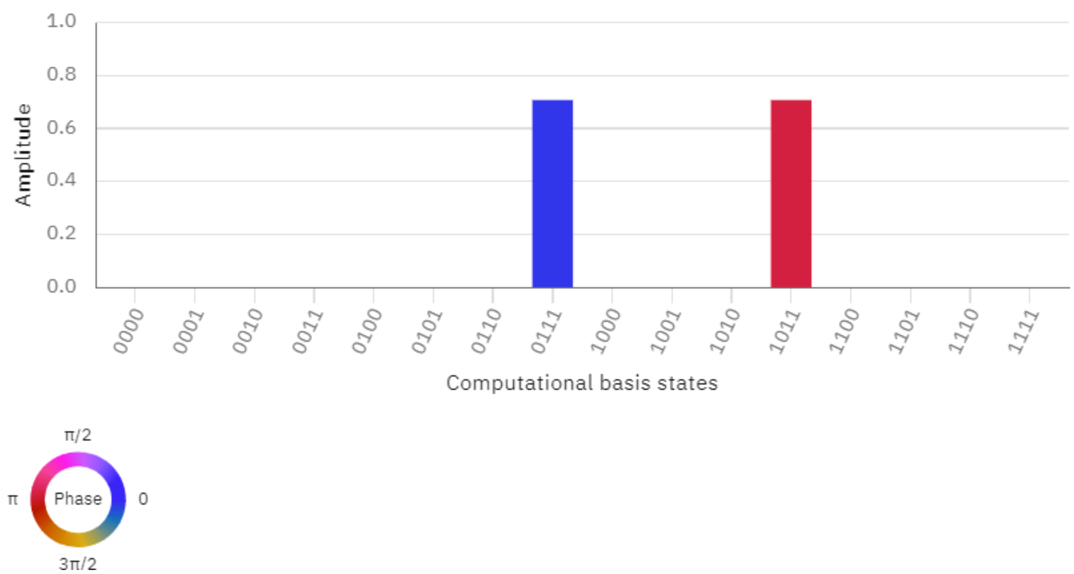


In the next step , apply CNOT gates to (q[0],q[3]) and (q[1],q[3])

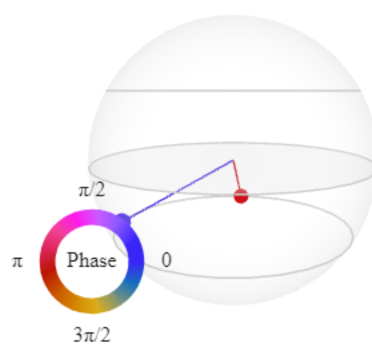
Results: The added gates affect the resulting qubits by [0+0j, 0+0j, 0+0j, 0+0j, 0.5+0j, 0+0j, 0+0j, -0.5+0j, 0+0j, 0.5+0j, -0.5+0j, 0+0j, 0+0j, 0+0j, 0+0j, 0+0j] In the upcoming step, apply H-gate on q[0] and q[1] and a CNOT gate and finally measure the q[0] to obtain the final result.



Results: Obtaining 16 qubits of states [0+0j, 0+0j, 0+0j, 0+0j, 0+0j, 0+0j, 0+0j, 0.707+0j, 0+0j, 0+0j, 0+0j, -0.707+0j, 0+0j, 0+0j, 0+0j, 0+0j] . Here is the bar graph illustrating the probability of each state.



And the Bloch Sphere representation



In the final state, we notice the creation of a superposition state . This experiment shed lights on the behavior of each gate and how it affects the state of a qubit with respect to other previously-applied gates.

7 : Quantum Noise in Quantum Computers

Quantum systems are open systems where they exchange energy and information with the environment, accordingly, the quantum systems are extremely sensitive to the interaction with the environment where this interaction is controlled and this is fundamental for extracting and processing information. When this interaction is not controlled, Quantum noise becomes significant. Generally, quantum noise refers to the effect of the environment on the qubit state introducing errors that are disruptive for the fidelity of information-processing protocols. While software bugs affect codes in quantum computers, the quantum errors result from the quantum hardware. Examples of errors arising from quantum noise include:

- **Bit-Flip Errors:**These are the errors causing a qubit to flip from "0" state to "1" state, or vice versa. This corresponds with a rotation about the X-axis of the Bloch sphere. In this situation, mitigation strategies aim to detect and reverse these flips.
- **Phase Errors:**These errors cause a qubit to rotate around the Z-axis of the Bloch sphere. Error mitigation techniques can be applied to correct these phase shifts and bring the qubit back to its intended position.
- **Decoherence:** This is a type of error that causes a qubit to lose its quantum properties and move towards a mixed state, represented by a point inside the Bloch sphere. Mitigation techniques try to preserve coherence for as long as possible. Decoherence could come from many aspects of the environment: changing magnetic and electric fields, radiation from warm objects nearby, or cross talk between qubits.
- **Gate error**arises from the imperfect implementation of quantum gates.

7.1 Quantum Noise Mitigation

Some techniques are applied to reduce the impact of noise on quantum systems to prevent occurrence of errors. These strategies include:

- **Fault–Tolerant Quantum Computers:**refers to designing quantum algorithms and circuits that can tolerate and correct errors through computational processes.
- **Dynamical Decoupling:**Includes applying sequences of pulses to qubit to cancel out the effect of noise.
- **Noise–Resilient Quantum Algorithms:**involves developing quantum algorithms that are less sensitive to certain types of noise or can operate with higher levels of noise.
- **Hardware Improvement:** it can be achieved through enhancing the quality and stability of qubits and strengthening their isolation of qubits from environmental interference.

7.2 The Relation between Quantum Error Mitigation and Qubit's State Position on a Bloch Sphere

The position of a state on the Bloch sphere resembles a great and precise representation of this state , which facilitates error mitigation as the detection of unintended changes caused by some kind of error becomes easier .

- **Phase Errors:** These errors cause a qubit to rotate around the Z-axis of the Bloch sphere. Error mitigation techniques can be applied to correct these phase shifts and bring the qubit back to its intended position.
- **Bit-Flip Errors:**These are the errors causing a qubit to flip from "0" state to "1" state, or vice versa. This corresponds with a rotation about the X-axis of the Bloch sphere. In this situation , mitigation strategies aim to detect and reverse these flips.
- **Decoherence:** This is a type of error that causes a qubit to lose its quantum properties and move towards a mixed state, represented by a point inside the Bloch sphere. Mitigation techniques try to preserve coherence for as long as possible.Decoherence could come from many aspects of the environment: changing magnetic and electric fields, radiation from warm objects nearby, or cross talk between qubits.
- **Calibration:** Quantum gates are realized as rotations on the Bloch sphere. There are calibration errors which cause over-rotations or under-rotations. To do error mitigation, corrections are made in these rotations to obtain the right end state. Essentially, quantum error mitigation has to do with ensuring that the position of a qubit on a Bloch sphere truly represents the intended state, in spite of errors.

8 Conclusion

When the first world quantum computer, Sycamore, was found, Google claimed its ability to compute faster than the fastest supercomputer, summit, by approximately 10000 years! Not only the higher computational speed but also the ability of quantum computers to solve complex problems is what makes it better than classical one. Despite the technical challenges, the future of quantum computing holds significant promise and potential across various fields and with continued research, transformative advances in quantum computing capabilities over the coming decades will occur.

9 References

1. Max Werninghaus, Walther,Meißner-Institute, How to control a qubit?
2. Oxford Ionics ,Controlling Qubits at Scale,
3. Ciaran Hughes . Joshua Isaacson. Anastasia Perry . Ranbel F.Sun and Jessica Turner, Quantum Computing of the Quantum Curious
4. Ilie Parvu, Gabriel Sandu, Iulian D.Toader , Romanian Studies in Philosophy of Science (Chapter 11)
5. CHI-KWONG LI, REBECCA ROBERTS, and XIAOYAN YIN , Decomposition of Unitary Matrices and Quantum Gates
6. The Bloch Sphere , Ian Glendinning , EUropean Centre for Parallel Computing at Vienna
7. MitiQ , What is Quantum Error Mitigation?