Quantum Technology

Justin H. Wilson

2024-12-21

Table of contents

Preface

The goal for this text is to be a snapshot in time of quantum technologies: the good, the bad, and the ugly. As of 2024, there has been a large amount of industry interest and progress to develop quantum technologies, and a student approaching this industry should have the basic knowledge to understand *what* is trying to be achieved and *how* they are trying to achieve it. To cut through the PR of industry, this text offers the author's personal perspective on what has been achieved, where things need to go, and the challenges to get there. This is not meant to be an authoritative guide on any one of the technologies presented here, but a jumping off point for the interested student.

Finally, this is a Quarto book.

To learn more about Quarto books visit https://quarto.org/docs/books.

1 Introduction

In 2019 [1], Google reached a significant milestone in quantum computing when they achieved "Quantum Advantage"–demonstrating for the first time that a quantum computer could surpass the capabilities of classical computers, albeit for a specific, specialized task. Using their Sycamore processor with 53 qubits, they showed that sampling from a quantum circuit a million times took only 200 seconds, while the equivalent task would take approximately 10,000 years on a classical supercomputer. This breakthrough marked a turning point in quantum computing, igniting increased interest and investment in quantum technologies. Building on this success, in December 2024, Google announced their new Willow chip [2], representing their latest advancement in quantum hardware. Today, numerous companies and research institutions worldwide are racing to develop quantum hardware, each pursuing different technological approaches to challenge the computational limits of classical computers.

> ℹ What is Quantum Advantage?
>
> Quantum advantage (or quantum supremacy) refers to the demonstration that a quantum computer can solve a specific problem significantly faster than the best known classical algorithm. However, the problem doesn't necessarily need to be useful–it just needs to be well-defined and verifiable.

This text aims to give undergraduate students a comprehensive introduction to quantum technology and computation. We will begin by exploring the fundamental mathematical framework that underlies quantum computing, building from basic principles to more advanced concepts. Crucially, we will see what specific things a quantum computer can achieve that a classical computer would struggle with. With this foundation, we will examine three of the most promising current quantum computing technologies: superconducting qubits, photonic quantum computing, and ion traps. Each of these approaches offers unique advantages and faces distinct challenges, which we will analyze in detail.

To bridge theory with practice, we will utilize IBM's Qiskit software platform to implement basic quantum computations, providing hands-on experience with quantum programming. As we progress, we will explore critical practical considerations in quantum computing, including error mitigation and correction strategies. Time permitting, we will venture into the cutting-edge field of topological quantum computing, which offers a potentially more robust approach to quantum computation.

Throughout this text, we will maintain a balanced perspective, examining both the tremendous potential and significant challenges facing quantum computing technology. Our goal is to equip students with both theoretical understanding and practical insights into this rapidly evolving field.

1.1 The Quantum-Classical Arms Race

The story of Google's quantum supremacy claim illustrates a fascinating dynamic in the field of quantum computing–an ongoing arms race between quantum and classical algorithms. When Google first announced their achievement with the Sycamore processor [1], they estimated that their quantum sampling task would take a classical supercomputer approximately 10,000 years. However, within months, IBM researchers developed improved classical algorithms that could potentially perform the same calculation in just 2.5 days [3]. Further work even demonstrated that using tensor networks, the problem could be solved *faster* on a modern superconductor with ExaFLOPS performance [4].

This back-and-forth highlights several important lessons. First, it demonstrates the remarkable adaptability of classical computing. As quantum computers advance, classical algorithm developers find increasingly clever ways to simulate quantum systems or solve specific problems more efficiently. This competition drives innovation in both fields–quantum hardware must continually improve to maintain its advantage, while classical algorithms become more sophisticated in response.

Second, it serves as a cautionary tale about interpreting quantum computing announcements, particularly those aimed at the general public. While the achievement of quantum advantage represents a genuine milestone, the initial 10,000-year estimate proved overly optimistic. This pattern has repeated with various quantum computing companies, where marketing claims sometimes outpace peer-reviewed scientific validation. For students and researchers in the field, it's crucial to maintain a balanced perspective–acknowledging genuine breakthroughs while critically evaluating bold claims.

The recent announcement of Google's Willow chip [2] represents another step forward, but should be viewed within this context of ongoing competition and careful validation. This healthy tension between quantum and classical approaches ultimately benefits both fields, pushing the boundaries of what's computationally possible while maintaining rigorous scientific standards.

1.2 Early Computing: The Lesson of Transistors

In Ref. [5] there are a few quotes about early computing

> *"Computers in the future may weigh no more than 1.5 tons."* –Popular Mechanics, forecasting the relentless march of science, 1949
>
> *"I think there is a world market for maybe five computers."* –Thomas Watson, chairman of IBM, 1943

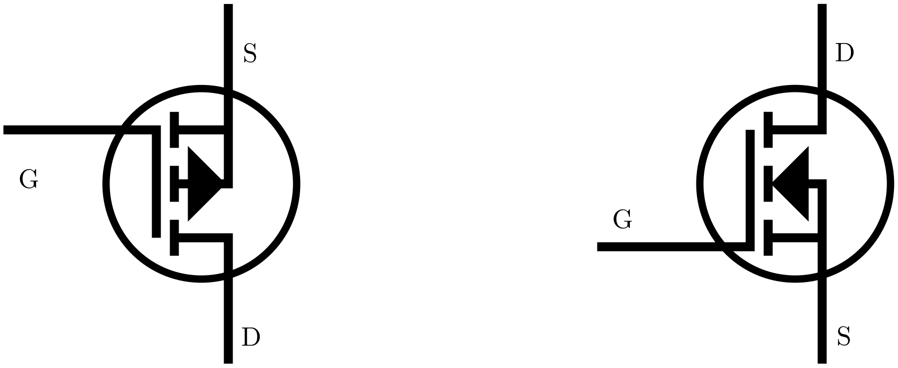
These quotes raise important points about early technology. While the theory of modern computing really took off with Turing in 1937 [6], the technological advancement necessary for modern computurs would not occur until later: when the transistor came about.

1.2.1 The Dream: Field Effect Transistors

To enable classical computing, we need something like a "switch" that can be on and off, keeping track of whether or not something like, current is flowing. This is hard to do with traditional circuit elements like resistors, capacitors, and inductors. It requires something more nonlinear: A switch that only allows current flow when a voltage is applied, a **transistor**.

Fig. 1.1 shows two of the types of circuit diagrams for a bipolar junction transistor (G = gate, S = source, and D = drain)[1]. A voltage applied at the gate enables a larger current to flow between collector and emitter. There are also *bipolar junction transistors* that use a smaller current to enable a larger current, in this case there is "base", "collector", and "emitter".

[1] This is also known as the CX-gate (the controlled-X gate).



(a) P-channel (b) N-channel

Fig. 1.1: Example Circuit diagrams for MOSFETs (*enh*)

With these building blocks, logical gates can be created, but *how* to build these? Which device can be made small and in abundance? And what challenges were encountered on the way.

Scale of Modern Computing in your pocket

The iPhone A17 Pro chip's 19 billion transistors would cover an area of about 1 square centimeter. If each transistor were the size of a grain of rice, they would cover an area larger than 75 football fields! This incredible miniaturization is what enables modern computing.

It was recognized early on that semiconductors provided an ideal platform for these kinds of circuits, and much work was done to try and create the above "field effect transistors." Schematically, these take the form:

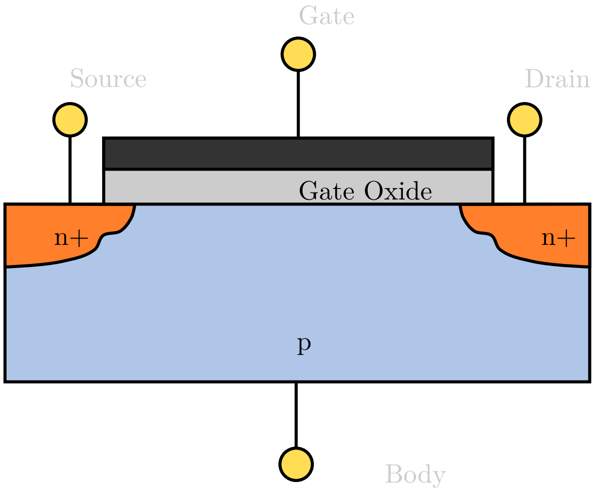
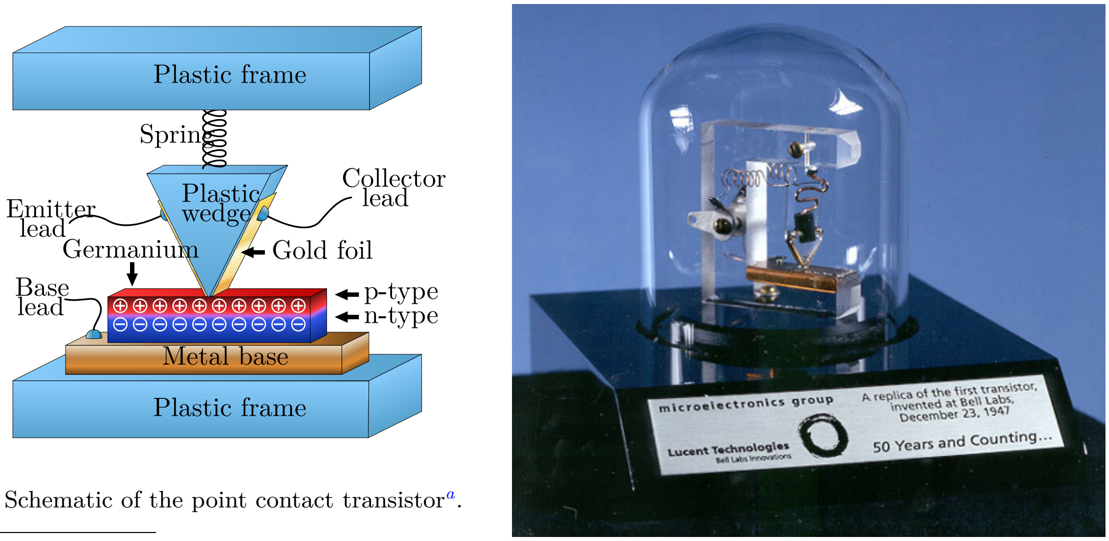


Fig. 1.2: Schematic of Field Effect Transistor[2]

[2] This is true for pure states, but can be easily generalized to mixed states, see [7].

In Fig. 1.2, electrons flow from source to drain, but only when the “gate” has an applied voltage to it (effectively “lowering the barrier” for electrons to get through). This theory was sound and based on the recently developed quantum electron theory of metals developed by Wolfgang Pauli, Werner Heisenberg, Arnold Sommerfeld, Felix Bloch, and Rudolf Peierls [8]. And indeed, the people at Bell labs worked on this problem theoretically and experimentally for the beginning in the 1930s (for a full history, see [8]). Despite the strong foundations though, creating a field effect transistor turned out to be difficult, and in the process Brattain and Bardeen instead created the point-contact transistor.

1.2.2 The Point Contact Transistor



(a) Schematic of the point contact transistor[a].

[a] Licensed under Creative Commons CC0 1.0 Universal Public Domain Dedication.

(b) Replica of first transistor

Fig. 1.3: The point contact transistor.

The point-contact transistor, invented in 1947, worked quite differently from the field-effect design. Instead of using a voltage at a gate to control current flow, it used two very closely spaced metal contacts pressed against a semiconductor (typically germanium). One contact, called the emitter, would inject positive charge carriers (holes) into the semiconductor. The second contact, called the collector, would collect these carriers - but crucially, the amount of current that could flow through the collector could be controlled by small changes in the emitter current[3]. A schematic and image of a replica of the original device are illustrated in Fig. 1.3.

This amplification effect, where a small current controls a larger one, was revolutionary–though the exact physics behind it wasn’t fully understood at the time. The key was that the metal contacts created special regions in the semiconductor where the positive carriers modified the barrier for current flow from the bulk material. While the detailed quantum mechanics is complex, you can think of it like creating “paths” that electrons prefer to take through the material, with the emitter current controlling how easily electrons can flow along these paths to the collector.

[3] For details on how this was made, see How the first transistor worked

While point-contact transistors were eventually superseded by more reliable and easier-to-manufacture designs, they represented a crucial breakthrough in electronics. They proved that solid-state devices could indeed amplify electrical signals. However, they were quite large. The original design for a field effect transistor would be needed, and they key resided in understanding and control the *surface physics* of semiconductors.

1.2.3 Surface physics and transistors

The key challenge in creating field effect transistors lay in understanding and controlling the surface properties of semiconductors. To understand why this was so difficult, let's break it down:

When a semiconductor crystal (like silicon) ends at a surface, something interesting happens. The regular pattern of atoms is suddenly interrupted - imagine a neat stack of blocks suddenly ending in mid-air. This interruption creates what we call "surface states" - special energy levels that electrons can occupy right at the surface of the material.

These surface states turned out to be extremely problematic for making transistors. Remember that in a field effect transistor, we want to control the flow of electrons using an electric field from the gate (see Fig. 1.2). However, these surface states acted like tiny electron traps, capturing and holding onto electrons. When electrons got stuck in these states, they effectively "screened" or blocked the electric field from the gate, preventing it from controlling the current flow through the semiconductor.

This screening effect was so strong that early attempts at field effect transistors simply didn't work–no matter how strong a voltage was applied to the gate, it couldn't effectively control the current flow. It was like trying to control a water flow with a valve, but having something constantly blocking the valve from moving.

The breakthrough came in the 1950s when researchers, particularly at Bell Labs, realized they needed to chemically "passivate" the semiconductor surface–essentially finding ways to neutralize these problematic surface states. The key discovery was that growing a thin layer of silicon dioxide (SiO) on silicon created a much more stable interface with far fewer problematic surface states. This oxide layer also served as an excellent insulator between the gate and the semiconductor.

This seemingly simple solution–growing an oxide layer–was actually a remarkable achievement that required precise control of material chemistry and manufacturing processes. It finally allowed the creation of practical field effect transistors, leading to the modern MOSFET (Metal-Oxide-Semiconductor Field Effect Transistor) that forms the backbone of today's electronics.

The success of this approach also highlights an important lesson in technology development: sometimes the biggest breakthroughs come not from changing the fundamental design, but from finding ways to control and manage the subtle physical effects that prevent a good design from working in practice.

1.2.4 Where are we with quantum computing?

Imagine that this course existed back in the 1950s and we called it "Computing Technology." Transistors were still coming online to enable computation at scale and we had both the information science and material theory to achieve it:

1. We knew what was needed to do universal classical computation.

2. We had the quantum theory of metals to describe how to build components (transistors) to achieve classical computation.

However, the engineering challenges took decades to resolve. It was only when we resolved those that computation as we currently envision it took off and we could have classical computers.

For quantum computing, we are in a similar situation:

1. We know what is needed to do universal quantum computation.
2. We have the quantum theory of photons, atoms, and superconductivity to achieve quantum computation.

However, as we will see in what follows, we have significant engineering challenges to achieve these in practice. While we will be largely concerned with the physics that make #2 possible in this course (and we'll touch on #1 for the first part of the course), we will pay attention to the strengths and weaknesses in the physics that lead to more pressing engineering challenges.

ℹ Historical Parallel

Just as the theory of classical computation [6] preceded practical computers by decades, we now have the theory of quantum computation [5] but face significant engineering challenges. The key difference is that we're trying to control individual quantum systems rather than classical electrical currents.

1.3 Review of the Postulates of Quantum Mechanics

Quantum mechanics is built out of some basic postulates which are crucial to understand for quantum computation. When we talk about a "system" in these postulates, we will be thinking of two-level systems which we can label 0 or 1 (for our qubit). However, we will state them generally since many applications require some basic work to reduce the complicated system down to just the qubits we are interested in.

❗ Mathematical Notation Guide

Throughout this text, we'll use:

- $|\psi\rangle$ ("ket psi"): Represents a quantum state
- $\langle\phi|$ ("bra phi"): The dual vector to $|\phi\rangle$
- $\langle\phi|\psi\rangle$: The inner product between states
- $\otimes$: The tensor product operation

These notations provide a compact way to describe quantum systems.

1.3.1 Postulate I: The Hilbert space

> A state in an isolated physical system S can be described by a set of *normalized state vectors*–$|\psi\rangle$ and all vectors related by a phase $|\psi'\rangle = e^{i\phi}|\psi\rangle$–belonging in the Hilbert space

$\mathcal{H}_S$.

importantly, a Hilbert space is equipped with an inner product (much like the dot product in three-dimensions) $\langle\alpha|\beta\rangle$.

We also need rules for attaching Hilbert spaces to one another. Afterall, we will need to use more than one qubit to do anything interesting.

> The Hilbert space of a composite system is the *tensor product* of the two individual systems $\mathcal{H}_{AB} = \mathcal{H}_A \otimes \mathcal{H}_B$.

1.3.2 Postulate II: Physical Observables and Measurements

This postulate is also sometimes called the **Born rule**.

In order to measure the system (e.g., "where is the particle?" or "Is the qubit a 0 or 1?"), we need to know what *physical observables* are

> (i) Every physical observable a can be described as a Hermitian operator A acting in the Hilbert Space.

Formally, a Hermitian operator has $A = A^\dagger$ where $A^\dagger$ is the *conjugate transpose* of A. These operators have a whole set of orthonormal eigenstates $A\,|a_n\rangle = a_n\,|a_n\rangle$ for a *real* number a_n (orthonormal means $\langle a_n|a_m\rangle = \delta_{nm}$). These mathematical details are important for how we will perform measurements

> (ii) When a physical observable with operator A is measured on a normalized eigenstate $|\psi\rangle$, the result is an eigenvalue a_n of that operator with probability $p_n = |\langle a_n|\psi\rangle|^2$ (or in the case of a degeneracy d, $p_n = \sum_{i=1}^{d} |\langle a_n, i|\psi\rangle\,|^2$).

If we measure A repeatedly, we are naturally lead to the expectation value $\langle\psi|A|\psi\rangle = a = \sum_n a_n p_n$, the average result of repeated quantum mechanical measurements. Once a measurement is performed, however, the state is changed, for this we need an operator P_n which projects onto the eigenspace of A associated with the eigenvalue a_n.

> (iii) When a measurement of the observable A gives a results a_n, the state is changed to be the *normalized projection* of $|\psi\rangle$ to the eigenspace associated with a_n

$$|\psi\rangle \quad \implies \quad \frac{P_n\,|\psi\rangle}{\sqrt{\langle\psi|P_n|\psi\rangle}}$$

If the eigenstates are not degenerate, then $|\psi\rangle \implies |a_n\rangle$. However, we will find that we will often have degenerate states, and in that case

$$|\psi\rangle \implies \frac{\sum_{i=1}^{d} \langle a_n, i|\psi\rangle\,|a_n, i\rangle}{\sqrt{\sum_{i=1}^{d} |\langle a_n, i|\psi\rangle|^2}}.$$

One comfortable with the braket notation might notice that within this, $P_n = \sum_{i=1}^{d} |a_n, i\rangle\,\langle a_n, i|$.

1.3.3 Postulate III: Time-evolution of a system

While we will talk about Hamiltonians, often in quantum computation we have gates that do not require these. In this case, we state this postulate as abstractly as possible

> The time evolution of a closed system from some time t_0 and state $|\psi_0\rangle$ to a final time t and state $|\psi\rangle$ can be described as a unitary transformation
>
> $$|\psi\rangle = U(t, t_0)\,|\psi_0\rangle\,.$$

This postulate is necessary for us to maintain probabilities. Unitary operators have the property that $UU^\dagger = U^\dagger U = \mathbb{1}$, and so

$$1 = \langle\psi_0|\psi_0\rangle = \langle\psi_0|U^\dagger U|\psi_0\rangle = \langle\psi|\psi\rangle\,.$$

In the case of time-independent *Hamiltonian dynamics,* the operator takes the form $U = e^{-iHt/\hbar}$ for a hermitian energy operator H called the *Hamiltonian.*

1.4 A Brief History of Computing: From Classical to Quantum

The concept that information is physical underlies both classical and quantum computation. Even in the earliest systems of record-keeping, we see physical objects encoding information:

- Kish Tablet (3500 BCE): A limestone tablet from Kish showing a record of pictographic writing.
- Quipu (2600–1900 BCE): A system of knotted ropes used by the Inca civilization for keeping records. The color, order, and number of knots all represented quantifiable or categorical data.

These examples highlight that from the very beginning, the act of storing and manipulating information has always had a physical basis—though the underlying physics often remained implicit for centuries.

1.4.1 Classical Computing Foundations

The paradigm of classical computing rests on a few fundamental ideas:

- **Boolean Logic & Universal Gates**: All classical computers can be built from a finite set of universal logical gates (e.g., {NAND}, {NOR}, or {AND, NOT}).

Most Boolean operations (AND, OR, NAND, etc.) are inherently irreversible: once a bit is erased or overwritten, the original state cannot be recovered from the output alone.

- **Extended (Physical) Church–Turing Thesis**: A probabilistic Turing machine can efficiently simulate any realistic physical model of computation with at most polynomial overhead.

This thesis, while unproven, underpins the expectation that classical computers (or at least classical models) are sufficient for simulating any physical system in principle. Quantum computation potentially challenges this with a polynomial in time algorithm (Shor's algorithm) that is exponential in time classically.

1.4.2 The Emergence of Quantum Information

While classical computing relies on bits that are strictly 0 or 1, quantum computing introduces powerful new concepts:

- **Superposition**: A quantum bit (qubit) can be in a linear combination of basis states (e.g., simultaneously "0" and "1" with certain complex amplitudes).
- **Entanglement**: Two or more qubits can become correlated in such a way that measuring one affects the outcomes for the others, even across vast distances.

The key question that launched the field of quantum information was whether these uniquely quantum properties—superposition and entanglement—could be exploited to perform computations more efficiently than any classical device.

In his seminal work, Simulating Physics with Computers [9], Richard Feynman observed that simulating quantum many-body systems on classical computers seems to require exponential resources. He posed the idea of harnessing genuine quantum systems themselves for simulation, planting the seeds for quantum computation as a research field.

Then, in a series of groundbreaking results between the 1980s and 1990s, researchers demonstrated that quantum computers could, in principle, outperform classical computers for certain tasks:

1. Deutsch (1985) [10]: Showed that it is possible to carry out a simple computational task on a quantum computer faster than any classical algorithm.
2. Deutsch–Jozsa (1992) [11]: Introduced a deterministic quantum algorithm that is exponentially faster (in the worst case) than any deterministic classical algorithm.
3. Bernstein–Vazirani (1992) [12]: Demonstrated a probabilistic quantum algorithm faster than any probabilistic classical algorithm.
4. Simon (1994) [13]: Provided a probabilistic quantum algorithm that is exponentially faster than any probabilistic classical algorithm for a specific promise problem.
5. Shor (1994) [14]: Showed how to factor integers efficiently, providing an exponential speedup over the best known classical methods. This result was particularly striking for cryptography, as factoring large numbers underpins many encryption schemes.

Alongside these algorithmic milestones, researchers began to delineate the limitations: not every problem can be exponentially sped up by quantum methods. For instance, Grover's algorithm (1996) [15] for unstructured search yields a quadratic speedup (from N to $\sqrt{N}$)—still better than classical, but not the exponential leap that Shor's algorithm provides for factoring.

1.4.3 Analog vs. Digital Quantum Simulation

As the field grew, quantum simulation branched into two approaches:

- Analog Quantum Simulation: Uses a controllable quantum system to mimic a target quantum system. The interactions in the simulator closely resemble the interactions in the system of interest.
- Digital Quantum Simulation: Decomposes a quantum evolution into a sequence of discrete gates (a "universal" set of quantum gates), akin to how classical digital computers function using logical gate operations.

Both approaches aim to exploit quantum mechanics to tackle problems in mathematics, physics, chemistry, and materials science that remain intractable for classical supercomputers.

Amid ongoing research, the interplay between classical and quantum paradigms remains a vibrant area of exploration. While classical computing infrastructure continues to be indispensable, quantum computing offers the promise of qualitatively new capabilities—provided we can tame the noise, errors, and fragilities inherent to quantum states.

1.5 Quantum Technologies

One major difference between the hindsight-history we have for classical computation and the current state of quantum computers is that are many platforms vying to enable quantum computation. There are arguments for and against each platform, and even arguments for using a combination of platforms. Here we give a list of some of the technologies and highlight the ones we will be surveying in this course.

Platforms covered in this course

- **Superconducting qubits**: Artificial atoms made from superconducting circuits that operate at ultra-low temperatures. Currently the most mature platform, used by companies like IBM and Google.
- **Trapped ions**: Individual atoms held in place by electromagnetic fields. Known for having very long coherence times and high-fidelity gates. Major players include IonQ and Quantinuum (formerly of Honeywell).
- **Photonic quantum computers**: Use particles of light (photons) as qubits. Can operate at room temperature and naturally interface with quantum communication systems. Being developed by companies like PsiQuantum and Xanadu.

Other platforms we will not cover

- **Silicon quantum dots**: Quantum bits made from individual electrons trapped in silicon, similar to classical semiconductor technology. Could potentially leverage existing manufacturing processes.
- **Neutral atoms and Rydberg arrays**: Individual neutral atoms arranged in arrays using laser beams. When excited to high-energy Rydberg states, atoms can interact strongly with their neighbors. Can create large numbers of identical qubits with programmable interactions. Companies like QuEra are pursuing this approach.
- **NV centers**: Quantum bits made from nitrogen-vacancy defects in diamond. Can operate at room temperature and have long coherence times, making them particularly promising for quantum sensing and networking applications.

A platform we will cover if time permits

- **Topological qubits**: A theoretical approach that would use special quantum states of matter to create error-protected qubits. Still in early research stages but could offer significant advantages if realized.

⚠ Current State of Quantum Computing

As of 2024, the largest quantum computers have around 50-100 physical qubits optimistically, but these are noisy and require error correction. For comparison, your smartphone has billions of classical bits. This highlights the early stage of quantum computing development and the engineering challenges ahead.

Each platform has its own advantages and challenges in terms of scalability, error rates, coherence times, and manufacturing complexity. The field is still evolving, and it's possible that different platforms may be optimal for different applications.

2 The Qubit

The fundamental building block of the classical computer was the bit: A 0 or 1 that could be manipulated by a classical computer (via transistors, see Section 1.2). In a similar manner, quantum computation has the "quantum bit" or just *qubit*, for short. This leads us to the linear algebra of 2 × 2 matrices, as we will see. Despite the apparent simplicity, we can already see many of the key features of quantum mechanics in this simple system.

⚠ Common Qubit Misconceptions

- A qubit is not just a probabilistic classical bit.
- You cannot directly access of the amplitudes of $|0\rangle$ and $|1\rangle$ through a single measurement.
- Superposition states collapse upon measurement.
- No-cloning theorem [16] means you cannot perfectly copy an unknown quantum state.
- Entanglement is not the same as classical correlation (see next chapter).

2.1 The Qubit Hilbert space

A qubit will be in two-dimensional complex vector space equipped with an inner product.

2.1.1 Qubit states

For the qubit, we associate two states with two different basis vectors: $|0\rangle$ and $|1\rangle$. This will be called the *computational basis*. The magic[1] of quantum mechanics is that a state need not be just one or the other, but could be *any* linear superposition of these

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle\,. \tag{2.1}$$

In this, we have adopted the *bra-ket* notation due to Dirac. While it can be quite useful, we can write this in terms of matrices and vectors

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

This makes it clear that these states are *orthogonal* $\langle 0|1\rangle = 0$.

In this case we have

$$|\psi\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}.$$

[1] Magic is a technical term in quantum computing, though we're using it in a colloquial sense here, see [17].

We also need the conjugate transpose, the Hermitian conjugate, of this vector, which will be a row-vector

$$\langle\psi| = \begin{bmatrix}\alpha^* & \beta^*\end{bmatrix}.$$

The key feature of quantum mechanics is that these states must be *normalized*, meaning that the probability of finding the system in any state must sum to 1. This means that

$$\langle\psi|\psi\rangle = \begin{bmatrix}\alpha^* & \beta^*\end{bmatrix}\begin{bmatrix}\alpha \\ \beta\end{bmatrix} = |\alpha|^2 + |\beta|^2 = 1.$$

In matrix notation, this is just the dot product of a vector with its complex conjugate.

ℹ Why Normalization Matters

The normalization condition $|\alpha|^2 + |\beta|^2 = 1$ isn't just mathematical convenience - it ensures probabilities add up to 100%! For example:

- $|\psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ gives 50-50 chance of measuring 0 or 1
- $|\psi\rangle = \frac{\sqrt{3}}{2}|0\rangle + \frac{1}{2}|1\rangle$ gives 75% chance of 0 and 25% chance of 1

We can also write operators that act on these states. The simplest operator is the Pauli Z operator, which in matrix form is

$$Z = \begin{bmatrix}1 & 0 \\ 0 & -1\end{bmatrix}.$$

When this operator acts on our basis states, we find

$$Z|0\rangle = |0\rangle, \quad Z|1\rangle = -|1\rangle.$$

This means that $|0\rangle$ and $|1\rangle$ are *eigenstates* of Z with eigenvalues $+1$ and -1 respectively. For a general state $|\psi\rangle$, measuring Z will yield either $+1$ or -1, with probabilities determined by $|\alpha|^2$ and $|\beta|^2$ respectively.

Example: Measuring a superposition state

Consider the state $|\psi\rangle = \frac{3}{5}|0\rangle + \frac{4}{5}|1\rangle$. When we measure this state in the Z basis:
Notice that this state is normalized since $(\frac{3}{5})^2 + (\frac{4}{5})^2 = 1$. If we measure this state in the computational basis:

- We'll get outcome $|0\rangle$ with probability $|\frac{3}{5}|^2 = 0.36$ (36%)
- We'll get outcome $|1\rangle$ with probability $|\frac{4}{5}|^2 = 0.64$ (64%)

After measurement, the state will collapse to either $|0\rangle$ or $|1\rangle$ with the above probabilities

⚠ Measurement Collapse in Practice

When we say a quantum state "collapses" upon measurement, what actually happens in the lab?

- For a superconducting qubit: We measure a voltage or current
- For an ion trap: We detect scattered photons

- For a photonic qubit: We count photons with a detector

Each technology has its own way of converting quantum information into classical signals!

2.1.2 Qubit operators

Since this is linear algebra, we can write a general operator $\mathcal{O}$ as a matrix

$$\mathcal{O} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}.$$

Often, we are interested in the eigenvalues and eigenstates of these operators $\mathcal{O} \left|\psi_i\right\rangle = \lambda_i \left|\psi_i\right\rangle$. Generically, we can find these by solving a polynomial equation

$$\det(\mathcal{O} - \lambda I) = 0.$$

Solving this step-by-step

$$\det(\mathcal{O} - \lambda I) = \begin{vmatrix} a-\lambda & b \\ c & d-\lambda \end{vmatrix} = 0,$$

which gives us

$$(a-\lambda)(d-\lambda) - bc = 0.$$

This is a quadratic equation that we can solve:

$$\lambda^2 - (a+d)\lambda + (ad - bc) = 0.$$

The eigenvalues are therefore

$$\lambda_\pm = \frac{a + d \pm \sqrt{(a-d)^2 + 4bc}}{2}. \tag{2.2}$$

For quantum mechanical **observables** (see Section 1.3.2), we are particularly interested in *Hermitian* operators where $\mathcal{O} = \mathcal{O}^\dagger$,

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} a^* & c^* \\ b^* & d^* \end{bmatrix}$$

This means that a and d must be real and $c = b^*$. In this case, the eigenvalues are always real, as we can see from the Eq. 2.2.

A particularly important class of operators are **unitary operators**, where $U^\dagger U = UU^\dagger = I$. These are what we use for time-evolution, see Section 1.3.3.

These operators preserve the inner product between states:

$$\langle U\psi | U\phi \rangle = \langle \psi | \phi \rangle$$

For a 2×2 matrix

$$U = \begin{bmatrix} a & b \\ c & d \end{bmatrix},$$

the unitarity condition means that

$$\begin{bmatrix} a^* & c^* \\ b^* & d^* \end{bmatrix} \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

This gives us several conditions:

1. $|a|^2 + |c|^2 = 1$ (normalization of first column)
2. $|b|^2 + |d|^2 = 1$ (normalization of second column)
3. $ab^* + cd^* = 0$ (orthogonality of columns)

This immediately gives us some insight into these operators. If we define,

$$|\psi_1\rangle = \begin{bmatrix} a \\ c \end{bmatrix}, \quad |\psi_2\rangle = \begin{bmatrix} b \\ d \end{bmatrix},$$

then we have $\langle\psi_1|\psi_1\rangle = 1 = \langle\psi_2|\psi_2\rangle$ and $\langle\psi_1|\psi_2\rangle = 0$.

An important property of unitary operators is that their eigenvalues always have magnitude 1, meaning they can be written as $e^{i\theta}$ for some real θ. This makes them natural operators for describing quantum evolution.

> Why Unitary?
>
> Unitary operators are special because they:
>
> 1. Preserve the normalization of quantum states
> 2. Are reversible (have an inverse)
> 3. Represent physical operations that conserve probability
>
> This is why quantum gates must be unitary - they represent real physical processes that can be undone!

2.1.3 The Pauli operators

A particularly important set of operators are the Pauli operators. We've already seen the Pauli Z operator. The other two are[2]

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}.$$

These operators satisfy some important algebraic relations:

$$X^2 = Y^2 = Z^2 = I, \quad XY = iZ, \quad YZ = iX, \quad ZX = iY.$$

We can additionally start to see some logical operations begin to appear; X operates on the computational basis as a **NOT** gate

$$X|0\rangle = |1\rangle, \quad X|1\rangle = |0\rangle.$$

[2] In some literature, these are matrices are denoted by σ_x, σ_y, and σ_z and related to spin operators via $S_i = \frac{1}{2}\sigma_i$. This insight can help bridge the idea of these operators and the Bloch sphere.

Pauli Operators in Action

The Pauli operators represent quantum operations:

- X is like the classical NOT gate: flips between $|0\rangle$ and $|1\rangle$
- Z adds a phase: leaves $|0\rangle$ alone but negates $|1\rangle$
- $Y = iXZ$ combines both operations.

These simple operations are building blocks for more complex quantum algorithms!

Example: Applying operators

Let's apply the X (NOT) gate to our state $|\psi\rangle = (\frac{3}{5}|0\rangle + \frac{4}{5}|1\rangle)$:

$$\begin{aligned} X|\psi\rangle &= X(\tfrac{3}{5}|0\rangle + \tfrac{4}{5}|1\rangle) \\ &= \tfrac{3}{5}X|0\rangle + \tfrac{4}{5}X|1\rangle \\ &= \tfrac{3}{5}|1\rangle + \tfrac{4}{5}|0\rangle \\ &= \tfrac{4}{5}|0\rangle + \tfrac{3}{5}|1\rangle \end{aligned}$$

The full set of Pauli operators, along with the identity, form a complete basis for 2 × 2 matrices, meaning we can write any operator as

$$\mathcal{O} = aI + bX + cY + dZ,$$

where a, b, c, and d are complex numbers. We can extract each of these numbers, mathematically, with a trace operation

$$\operatorname{tr}\mathcal{O} = 2a, \quad \operatorname{tr}\mathcal{O}X = 2b, \quad \operatorname{tr}\mathcal{O}Y = 2c, \quad \operatorname{tr}\mathcal{O}Z = 2d.$$

Note that separately, X, Y, and Z are Hermitian (and thus, observables). If $\mathcal{O}$ is an observable, then $\mathcal{O} = \mathcal{O}^\dagger$ immediately leads us to a, b, c, and d being all real.

We can put constraints on these coefficients for unitary operators as well, and we leave this as an exercise for the reader.

Finally, these operators have eigenstates as well, and we can define them as $X|\pm\rangle = \pm|\pm\rangle$ and $Y|\pm i\rangle = \pm|\pm i\rangle$, and they have the forms

$$\begin{aligned} |\pm\rangle &= \tfrac{1}{\sqrt{2}}(|0\rangle \pm |1\rangle), \\ |\pm i\rangle &= \tfrac{1}{\sqrt{2}}(|0\rangle \pm i|1\rangle). \end{aligned}$$

We will often want to change our basis from $|0\rangle$ and $|1\rangle$ to $|+\rangle$ and $|-\rangle$. This is accomplished with something called the *Hadamard gate* (we'll call it H, not to be confused with a Hamiltonian) and it is created specifically to change from computational basis to the X basis: $H|0\rangle = |+\rangle$ and $H|1\rangle = |-\rangle$. As a matrix it takes the form

$$H = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}.$$

! The Power of Hadamard

The Hadamard gate is one of the most important gates in quantum computing with useful properties:

1. It creates equal superpositions from computational basis states.
2. It's its own inverse ($H^2 = I$)
3. It's used in nearly every quantum algorithm
4. When applied to n qubits, it creates a superposition of all 2^n possible bit strings!

Example 3: The Hadamard Transform

The Hadamard gate is particularly important because it creates superposition states. Let's see what happens when we apply it to $|0\rangle$:

$$\begin{aligned} H\,|0\rangle &= \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\ &= \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \\ &= \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \\ &= |+\rangle \end{aligned}$$

2.2 The Bloch sphere

The qubit itself is more than just a probability of being a 1 or a 0. A crucial bit of *quantum* information is the relative phase between the two states for instance

$$|\psi\rangle = \tfrac{1}{\sqrt{2}}(|0\rangle + e^{i\phi}\,|1\rangle),$$

and since all states are equivalent up to a *total* phase, we can write the amplitude of each state with a real number. In this case, if we set $|\psi\rangle = x\,|0\rangle + ye^{i\phi}\,|1\rangle$, then we have $\langle\psi|\psi\rangle = x^2 + y^2 = 1$ for normalization. This is the equation for a circle, and writing out $x = \cos(\theta/2)$ and $y = \sin(\theta/2)$[3], we are lead to an angular representation of our state.

$$|\psi\rangle = \cos(\theta/2)\,|0\rangle + \sin(\theta/2)e^{i\phi}\,|1\rangle\,. \tag{2.3}$$

Let's see how this state relates to our Pauli operators. If we calculate the expectation values of each operator:

[3] The divide-by-two for the angles will become clear as we go through this section.

$$
\begin{aligned}
\langle X\rangle &= \langle\psi|\, X\, |\psi\rangle \\
&= (\cos(\theta/2)\, \langle 0| + \sin(\theta/2) e^{-i\phi}\, \langle 1|) X (\cos(\theta/2)\, |0\rangle + \sin(\theta/2) e^{i\phi}\, |1\rangle) \\
&= (\cos(\theta/2)\, \langle 0| + \sin(\theta/2) e^{-i\phi}\, \langle 1|)(\cos(\theta/2)\, |1\rangle + \sin(\theta/2) e^{i\phi}\, |0\rangle) \\
&= \sin(\theta/2)\cos(\theta/2) e^{i\phi} + \cos(\theta/2)\sin(\theta/2) e^{-i\phi} \\
&= \sin(\theta/2)\cos(\theta/2)(e^{i\phi} + e^{-i\phi}) \\
&= 2\sin(\theta/2)\cos(\theta/2)\cos\phi \\
&= \sin\theta\cos\phi.
\end{aligned}
$$

We can carry out a similar calculation for Y and Z to obtain

$$
\begin{aligned}
\langle X\rangle &= \langle\psi|\, X\, |\psi\rangle = \sin\theta\cos\phi \\
\langle Y\rangle &= \langle\psi|\, Y|\psi\rangle = \sin\theta\sin\phi \\
\langle Z\rangle &= \langle\psi|\, Z\, |\psi\rangle = \cos\theta
\end{aligned}
$$

These expectation values give us coordinates , which are precisely the coordinates of a point on a unit sphere! This is why we call it the Bloch sphere. The angles θ and ϕ are the usual spherical coordinates.

The Bloch Sphere Geometry

- The north pole ($\theta = 0$) corresponds to $|0\rangle$
- The south pole ($\theta = \pi$) corresponds to $|1\rangle$
- The equator ($\theta = \pi/2$) contains equal superposition of computational basis states:
  - $\phi = 0$ gives $|+\rangle$ (positive x-axis)
  - $\phi = \pi$ gives $|-\rangle$ (negative x-axis)
  - $\phi = \pi/2$ gives $|+i\rangle$ (positive y-axis)
  - $\phi = 3\pi/2$ gives $|-i\rangle$ (negative y-axis)

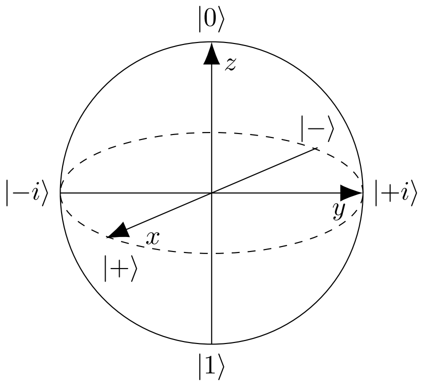


Fig. 2.1: The Bloch sphere showing eigenstates of X, Y, and Z Pauli operators

Example: Hadamard Gate on $|0\rangle$

The Hadamard gate H takes the state $|0\rangle$ (north pole) to $|+\rangle$ (on the equator at $\phi = 0$). In terms of the Bloch sphere coordinates, this means:

- Starting point: $\theta = 0$ (north pole)
- Ending point: $\theta = \pi/2$, $\phi = 0$ (positive x-axis)

The gate effectively rotates the state by 90° around the y-axis. Similarly, $H|1\rangle$ takes the south pole to $|-\rangle$ on the negative x-axis.

2.2.1 General Unitary Rotations

The Hadamard example shows how unitary gates can rotate states on the Bloch sphere. More generally, any single-qubit unitary operation can be thought of as a rotation of the Bloch sphere. Let's see how this works.

A general rotation around a unit vector $\vec{n} = (n_x, n_y, n_z)$ by angle θ is given by

$$R_{\vec{n}}(\theta) = \cos(\theta/2)I - i\sin(\theta/2)(n_xX + n_yY + n_zZ).$$

For example:

- Rotation around z-axis:
$$R_z(\theta) = e^{-i\theta Z/2} = \begin{bmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{bmatrix}$$
- Rotation around x-axis:
$$R_x(\theta) = e^{-i\theta X/2} = \begin{bmatrix} \cos(\theta/2) & -i\sin(\theta/2) \\ -i\sin(\theta/2) & \cos(\theta/2) \end{bmatrix}$$
- Rotation around y-axis:
$$R_y(\theta) = e^{-i\theta Y/2} = \begin{bmatrix} \cos(\theta/2) & -\sin(\theta/2) \\ \sin(\theta/2) & \cos(\theta/2) \end{bmatrix}$$

A remarkable fact is that any single-qubit unitary operation can be decomposed into three rotations around two different axes. This is known as the Euler angle decomposition:

$$U = e^{i\alpha}R_z(\phi)R_y(\theta)R_z(\psi)$$

where α, ϕ, θ, and ψ are real numbers. The global phase $e^{i\alpha}$ is often unimportant for quantum computing purposes.

Visualizing Rotations

The Euler angle decomposition has a nice geometric interpretation:

1. First rotation ($R_z(\psi)$): Rotate around z-axis

2. Second rotation ($R_y(\theta)$): Tilt to new latitude
3. Third rotation ($R_z(\phi)$): Rotate to final longitude
4. Global phase ($e^{i\alpha}$): Invisible in measurements

This is similar to how we specify points on Earth using latitude and longitude!

2.2.2 The Phase Gate

The phase gate (often denoted as S) is another important single-qubit gate that adds a phase of i to the $|1\rangle$ state while leaving $|0\rangle$ unchanged:

$$S = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}$$

When acting on basis states:

$$S|0\rangle = |0\rangle, \quad S|1\rangle = i|1\rangle$$

The phase gate is equivalent to a $\pi/2$ rotation around the z-axis: $S = R_z(\pi/2)$. On the Bloch sphere, this corresponds to rotating a state by 90° around the z-axis.

Example: Phase Gate on Superposition

Let's see what happens when we apply S to an equal superposition state:

$$\begin{aligned} S|+\rangle &= S\left(\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)\right) \\ &= \frac{1}{\sqrt{2}}(S|0\rangle + S|1\rangle) \\ &= \frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle) \end{aligned}$$

This transforms $|+\rangle$ into $|+i\rangle$, rotating it from the positive x-axis to the positive y-axis on the Bloch sphere.

The phase gate is particularly important in quantum error correction and quantum algorithms where controlled phase operations are needed.

More generically, we can create any $R_z(\phi)$ gate to perform rotations about the z-axis (this is very useful for Shor's algorithm). However, a common variant is the T gate, which is simply $R_z(\pi/4)$ and is one of the minimal components needed to achieve universal quantum computation.

! Phase gate leaves computational basis states alone

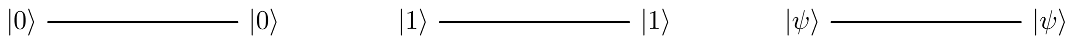
Note that these gates only change superposition of computational basis states, so $S|0\rangle = |0\rangle$ and $S|1\rangle = i|1\rangle$. (Similarly for any gate made from rotations about the z-axis.)

2.3 Quantum Circuits

Now that we've covered the key single-qubit operations, we can start to think about how to represent sequences of these operations graphically using quantum circuits. In quantum circuits:

- Qubits are represented as horizontal lines (called "wires", see Fig. 2.2).
- Gates are boxes or symbols placed on these wires (see Fig. 2.3 and Fig. 2.4).
- Time flows from left to right.
- Measurements are represented by meters (see Fig. 2.5).

Often, unless we are preparing a specific state for an algorithm, we may leave off the states from the ends of the "wire." This wire, Fig. 2.2, you can think of as the identity.



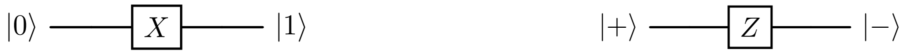
(a) Identity on $|0\rangle$ (b) Identity on $|1\rangle$ (c) Identity on $|\psi\rangle$

Fig. 2.2: Lines representing the evolution of a qubit (no gates applied)

While in quantum state evolution we apply operators right-to-left, the two operations $X|0\rangle = |1\rangle$ and $Z|+\rangle = |-\rangle$ are represented in Fig. 2.3.

! Important

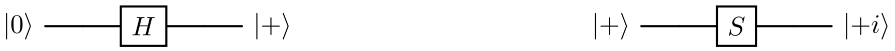
The Pauli operators are both unitary and Hermitian, so they perform as quantum gates (which require unitary as Section 1.3.3 stipulates) and as observables (as Section 1.3.2 stipulates).



(a) Application of the X gate. (b) Application of the Z gate.

Fig. 2.3: Examples of the Pauli operators as gates.

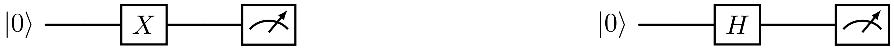
The Hadamard and phase gates can also be mixed in, Fig. 2.4, and we can even introduce the meter symbol for measurements, Fig. 2.5.



(a) Application of the Hadamard gate. (b) Application of the phase gate

Fig. 2.4: Examples of the Hadamard and phase gates.

Whenever measurements are performed, we will often transform to the computational basis $|0\rangle$ and $|1\rangle$ to perform a measurement. By playing tricks with unitary transformations, we can measure other observables by mixing unitaries and measurements of the computational basis; dependent on the platform, this may or may not be necessary. In terms of diagrams, you should assume meters are measurements in the computational basis unless it is noted otherwise.



(a) Will measure 1 with 100% probability

(b) Will measure 1 with 50% probability

Fig. 2.5: Examples of measurements. The meter is assumed to be measuring in the computational basis unless otherwise noted (i.e., measuring the Z operator).

Reading Quantum Circuits

Quantum circuits are read from left to right, just like reading text. Each horizontal line represents a qubit's journey through time, and the boxes show what operations happen and when.
The measurement symbol at the end indicates when we extract classical information from our quantum system.

These diagrams give us a powerful visual language for describing quantum computations. Even complex algorithms can be broken down into sequences of these basic operations.

Example: Gate Sequences

Consider applying H then Z then H to $|0\rangle$:

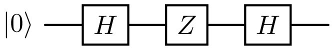


Fig. 2.6: Implementing an X gate with Z and H.

1. $H|0\rangle = |+\rangle$ (move to +x axis)
2. $Z|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$ (rotate around z by 90°)
3. $H(\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)) = |1\rangle$ (move to -z axis)

This sequence effectively implements the X gate!

2.4 Noise and decoherence

A large part of the course will consist of finding physical systems where we can identify certain states as $|0\rangle$ and $|1\rangle$, and then proceed to figure out how to perform the gates we need for quantum algorithms. In this way, this course could be called "two-level systems and where to find them." However, these states are not pristine and so we need a way to talk about states that subject to noisy environments and loss of information.

In fact, environmental interactions can cause, amongst other issues:

- Loss of phase information (dephasing)
- Loss of energy (amplitude damping)
- Random bit flips

To properly describe these noise processes, we need to move beyond pure state descriptions and introduce the density matrix formalism.

2.4.1 The Density Matrix

For a pure quantum state $|\psi\rangle$, the density matrix is defined as

$$\rho = |\psi\rangle\langle\psi|$$

For example, the computational basis states have density matrices:

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

The density matrix also provides a convenient way to calculate expectation values of observables. For any observable A, the expectation value is given by:

$$\langle A\rangle = \mathrm{tr}(A\rho)$$

For a pure state $|\psi\rangle$, this reduces to our familiar expression:

$$\mathrm{tr}(A|\psi\rangle\langle\psi|) = \langle\psi| A |\psi\rangle$$

Example: Measuring Z with the Density Matrix

Consider measuring Z for the state $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$:
The density matrix is:

$$\rho_+ = \frac{1}{2}\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

Using the formula for expecation values with $Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$:

$$\langle Z\rangle = \mathrm{tr}(Z\rho_+) = \mathrm{tr}\left(\frac{1}{2}\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}\right) = 0$$

This matches what we expect: $|+\rangle$ has equal probability of measuring ±1 for Z.

The real power of density matrices comes from describing *mixed states*, which are statistical mixtures of pure states:

$$\rho = \sum_i p_i |\psi_i\rangle\langle\psi_i|$$

where p_i are classical probabilities ($p_i \geq 0$, $\sum_i p_i = 1$).

i Properties of Density Matrices

Any valid density matrix must satisfy:

1. Hermiticity: $\rho = \rho^\dagger$

2. Positive semidefinite: $\langle\phi| \rho |\phi\rangle \geq 0$ for all $|\phi\rangle$
3. Unit trace: $\mathrm{tr}(\rho) = 1$
4. For pure states: $\mathrm{tr}(\rho^2) = 1$
5. For mixed states: $\mathrm{tr}(\rho^2) < 1$

Quantum Operations with Density Matrices

Just as we can evolve pure states with unitary operators, we can evolve density matrices. For a unitary operation U, the density matrix transforms as:

$$\rho \to U\rho U^\dagger$$

This preserves all the properties of the density matrix we discussed above.

When we make a measurement, we use projection operators $\{P_m\}$ (like $|0\rangle \langle 0|$ and $|1\rangle \langle 1|$ for measuring in the computational basis). The probability of getting outcome m is:

$$p(m) = \mathrm{tr}(P_m \rho)$$

After measuring and getting outcome m, the state "collapses" to:

$$\rho_m = \frac{P_m \rho P_m}{\mathrm{tr}(P_m \rho)}$$

where the denominator ensures the trace remains 1.

> ℹ Quantum Channels
>
> The density matrix formalism really shines when describing general quantum evolution, including noise. Any physical process (unitary or not) can be described by a quantum channel $\mathcal{E}$:
>
> $$\rho \to \mathcal{E}(\rho) = \sum_k E_k \rho E_k^\dagger$$
>
> The operators E_k (called Kraus operators) satisfy $\sum_k E_k^\dagger E_k = I$ to preserve probability. This includes both ideal unitary evolution (one $E_k = U$) and noise processes (multiple E_k), as we'll see in the next section.

Density Matrices and the Bloch Sphere

The density matrix formalism provides a beautiful geometric picture when combined with the Bloch sphere representation in Section 2.2. While pure states live on the surface of the Bloch sphere, mixed states live *inside* it. Any density matrix for a single qubit can be written as:

$$\rho = \frac{1}{2}(I + \vec{r} \cdot \vec{\sigma})$$

where $\vec{r} = (r_x, r_y, r_z)$ is called the Bloch vector and $\vec{\sigma} = (X, Y, Z)$ are the Pauli matrices.

The length of $\vec{r}$ determines how mixed the state is:

- For pure states, $|\vec{r}| = 1$ (surface of sphere)
- For mixed states, $|\vec{r}| < 1$ (inside sphere)
- For the maximally mixed state, $\vec{r} = 0$ (center of sphere)

This gives us an intuitive picture of decoherence: noise processes like dephasing and bit flips move the Bloch vector inward from the surface, representing the loss of quantum information. The maximally mixed state at the center represents complete loss of information–equal probabilities for all measurement outcomes.

Connection to Purity

The length of the Bloch vector is directly related to the purity $\mathrm{tr}(\rho^2)$ we discussed earlier:

$$\mathrm{tr}(\rho^2) = \frac{1 + |\vec{r}|^2}{2}$$

This confirms that pure states ($|\vec{r}| = 1$) have purity 1, while mixed states have purity less than 1.

2.4.2 Modeling Noise

A full discussion of different quantum channels can be found in [18]. Here, we discuss some of the ones relevant for loss of quantum information.

The language of density matrices allows us to describe some of the errors that can occur in a precise manner

Random bit-flips

If there is probability p that a bit is flipped, we can encode this within a change of the density matrix:

$$\rho \to (1-p)\rho + pX\rho X,$$

where the first term describes the probability that the density matrix remains unchanged and the second describe the process of randomly flipping a bit.

Dephasing

As we've already mentioned the phase between $|0\rangle$ and $|1\rangle$ is useful information that we will take advantage of. However, some physical processes can scramble this phase in a way that could be inherently difficult to parse, in that case we can describe these processes with "dephasing"

$$\rho \to (1-p)\rho + pZ\rho Z.$$

This process has a clear action on ρ when $p = 1/2$. To illustrate this, consider $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, this has a probability of measuring 0 of $|\alpha|^2$ and a probabilty of measuring 1 of $|\beta|^2$, but there is also phase information between α and β that tells us where on the Bloch sphere the state is located. Dephasing will eliminate this phase information.

To see this, the full density matrix before we apply dephasing is

$$\rho = \begin{bmatrix} |\alpha|^2 & \alpha\beta^* \\ \alpha^*\beta & |\beta|^2 \end{bmatrix},$$

and after dephasing we will have the density matrix

$$\frac{1}{2}\rho + \frac{1}{2}Z\rho Z = \begin{bmatrix} |\alpha|^2 & 0 \\ 0 & |\beta|^2 \end{bmatrix}.$$

The off-diagonal components carried the relative phase information and it is now entirely lost, leaving us with a classical mixture of $|0\rangle$ and $|1\rangle$

! Measurement as Dephasing

This process of losing phase information is exactly what happens when we measure a quantum state in a particular basis! When we measure in the computational basis ($\{|0\rangle, |1\rangle\}$), we are effectively performing complete dephasing - we destroy all phase information between the basis states and are left with only the classical probabilities. This is why measurement is often described as "collapsing" the quantum state into classical information.

This connection between measurement and dephasing illustrates a fundamental aspect of quantum mechanics: the act of gaining classical information about a quantum system necessarily destroys some of its quantum properties, as formalized in the measurement postulates we discussed earlier.

Example: Dephasing of a Superposition State

Consider the state $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ under dephasing:
Initial density matrix:

$$\rho_0 = \frac{1}{2}\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

After dephasing with probability p:

$$\rho(p) = \frac{1}{2}\begin{bmatrix} 1 & (1-2p) \\ (1-2p) & 1 \end{bmatrix}$$

Complete dephasing ($p = \frac{1}{2}$) yields the maximally mixed state:

$$\rho(\tfrac{1}{2}) = \frac{1}{2}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Depolarizing

The depolarizing channel represents one of the most severe forms of noise in quantum systems, often considered the "worst-case scenario" for quantum information. This channel transforms any input state into a mixture of itself and the maximally mixed state:

$$\rho \to (1-p)\rho + p\frac{I}{2},$$

where p represents the probability of depolarization. When $p = 1$, the state becomes completely mixed regardless of the input:

$$\rho \to \frac{1}{2}I.$$

Note that this has eliminated any information about $|\psi\rangle$!

Example: The Pauli Twirl

An interesting way to understand the depolarizing channel is through what's called the "Pauli twirl". The depolarizing channel can be written as a random application of Pauli operators:

$$\rho \to (1-p)\rho + \frac{p}{3}(X\rho X + Y\rho Y + Z\rho Z).$$

This means we can implement depolarizing noise by randomly applying X, Y, or Z gates with probability $p/3$ each. This is particularly useful in quantum error correction, where we often want to simulate noise in a way that's easy to analyze.

Amplitude Damping

In many physical systems, one can lose information by having emission of energy. Physically, this could be an atom in some energy level and it emits a photon, falling to a lower energy level. One can imagine a process like this occuring with an operator

$$E_0 = \sqrt{\gamma}\,|0\rangle\langle 1|,$$

which describes the process of taking an occupied state $|1\rangle$ and transforming it to $|0\rangle$ with probability γ. To fully encode this into a change in the density matrix, we need one other operator: the chance that nothing happens. A natural guess would be

$$E_1 = |0\rangle\langle 0| + \sqrt{1-\gamma}\,|1\rangle\langle 1|,$$

which describes that there is unit probability of remaining $|0\rangle$ and probaiblity $1-\gamma$ of remaining in $|1\rangle$ if you start there.

The full operation on the density matrix is then

$$\rho \to E_1 \rho E_1^\dagger + E_0 \rho E_0^\dagger$$

Example: Amplitude Damping in Action

Let's see how amplitude damping affects a simple superposition state. Consider the initial state

$$|\psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle),$$

which has density matrix

$$\rho = \frac{1}{2}\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}.$$

After applying amplitude damping with probability γ, the state becomes

$$\begin{aligned}\rho' &= E_1 \rho E_1^\dagger + E_0 \rho E_0^\dagger \\ &= \frac{1}{2}\begin{bmatrix} 1+\gamma & \sqrt{1-\gamma} \\ \sqrt{1-\gamma} & 1-\gamma \end{bmatrix}.\end{aligned}$$

We can see that:

- The probability of being in $|1\rangle$ decreases by γ
- The probability of being in $|0\rangle$ increases by γ
- The off-diagonal coherence terms decay by $\sqrt{1-\gamma}$

When $\gamma = 1$ (complete damping), the state becomes

$$\rho' = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} = |0\rangle \langle 0| \,,$$

representing complete relaxation to the ground state.

Erasure Channel

Another important quantum channel is the erasure channel, which models the loss of a qubit to other accessible states. Unlike amplitude damping, where information leaks gradually, the erasure channel represents a complete loss of the qubit with probability ϵ. When this happens, the qubit is replaced by an “error state” that we denote as $|e\rangle$. Importantly, $|e\rangle$ exists in a different part of the Hilbert space than our qubit states - it’s an additional state that flags that erasure has occurred.

! Orthogonality of Error State

The error state $|e\rangle$ is orthogonal to both $|0\rangle$ and $|1\rangle$, meaning $\langle e|0\rangle = \langle e|1\rangle = 0$. This is crucial as it represents a state completely outside our original qubit space but still within the system itself.

The erasure can then be represented by

$$\rho \mapsto (1-\epsilon)\rho + \epsilon \, |e\rangle \langle e| \,.$$

In particular, this process

- Preserves the state with probability $1-\epsilon$
- Erases it completely with probability ϵ, replacing it with the error state $|e\rangle$

Example: Erasure Channel in Action

Since the erasure channel operates in an enlarged Hilbert space that includes the error state (in this case, making it 3-dimensional), let’s consider our superposition state embedded in this expanded space:

$$|\psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\1\\0\end{bmatrix}.$$

After the erasure channel acts, the state becomes a mixed state:

$$\rho' = (1-\epsilon)\begin{bmatrix}1/2 & 1/2 & 0\\1/2 & 1/2 & 0\\0 & 0 & 0\end{bmatrix} + \epsilon\begin{bmatrix}0 & 0 & 0\\0 & 0 & 0\\0 & 0 & 1\end{bmatrix} = \begin{bmatrix}\frac{1-\epsilon}{2} & \frac{1-\epsilon}{2} & 0\\\frac{1-\epsilon}{2} & \frac{1-\epsilon}{2} & 0\\0 & 0 & \epsilon\end{bmatrix}$$

This represents that with probability $1-\epsilon$ we still have our original state in the upper-left 2×2 block, but with probability ϵ we have completely lost it to the error state, represented by the 1 in the bottom-right corner.

The erasure channel is particularly important in quantum communication and error correction because the error state exists within the system Hilbert space itself (not the environment). This additional state in the system allows us to track and flag when errors occur–making it especially useful for quantum error correction protocols, unlike other noise channels where the errors are unknown and harder to detect.

2.4.3 Physical Qubit Implementations

The quest for building practical quantum computers has led to several different approaches for implementing qubits. Each implementation aims to create a physical system that can reliably represent quantum states and allow for precise control and manipulation. We will explore in detail later what is needed for a good quantum device, but give a brief overview here of *what* the devices are.

Superconducting Qubits

- Qubit are energy levels of *charge* or *flux* within a superconducting circuit.
- Based on superconducting circuits using Josephson junctions.
- Operate at extremely low temperatures (~20 mK).
- Used by: IBM, Google, Rigetti.

Trapped Ion Qubits

- Qubits are electronic or nuclear states of individual ions
- Ions held in electromagnetic traps, manipulated by lasers
- Used by: IonQ, Honeywell-Quantinuum

Photonic Qubits

- Qubits are properites of light (e.g., polarization)
- Can operate at room temperature
- Used by: PsiQuantum, Xanadu

Semiconductor Quantum Dots

- Qubits can be encoded using either:
  - *charge* states (electron occupying different quantum dots)
  - *spin* states (up/down spin states of an electron)
- Created by confining electrons in semiconductor nanostructures
- Often called "artificial atoms" due to their discrete energy levels
- Active research at: Intel, TU Delft, Princeton, UNSW, CEA-Leti

NV Centers in Diamond

- The nitrogen vacancy defects in diamond provide energy levels accessible with light.
- Can operate at room temperature
- Applications in quantum sensing and networking

Topological Qubits

- Qubits are non-local and topologically protected states within an exotic (topological) state of matter.
- Most promising candidate: Majorana zero modes
- **Current Status**:
  - Active research at Microsoft and Delft
  - Recent progress in identifying Majorana signatures in nanowires
  - Debate continues over experimental evidence

As we'll discuss, the "perfect" qubit would combine long coherence times, fast gates, high fidelity, easy coupling to other qubits, and straightforward scalability. While each implementation has made significant progress, achieving all these properties simultaneously remains a major challenge in the field.

3 Multiple Qubits

Previously in Chapter 2 we discussed in detail how to understand a single qubit. While we saw some basic features, such as superposition and relative phase, it has not been apparent yet what we can do with these features. The power of these will really be unlocked by putting multiple qubits together. We will also begin to see hints of *quantum entanglement*.

However, there is just a practical concern: How much information can we store in a single qubit? With precise control, we have our answer in Section 2.2: the angles θ and ϕ on the Bloch sphere. Since algorithms are bit more greedy than that, we need to extend our space and the natural way to do that is to add more qubits. Of course, classically, we operate with many bits: Floats (real numbers) on most computers use 64 bits, and we often add, substract, multiply a lot of these numbers. But classically, when we have, for instance, two bits, there are four discrete states 00, 01, 10, and 11. As we will analyze in detail, quantum mechanically these will be four basis states that can make up a general quantum wave function

$$|\psi\rangle = \alpha\,|00\rangle + \beta\,|01\rangle + \gamma\,|10\rangle + \delta\,|11\rangle\,. \tag{3.1}$$

3.1 Tensor Products and the multi-qubit Hilbert space

The equation above gives us an idea for how we ought to combine qubits. The tensor product will formalize this, but we will build it up “intuitively.”

Imagine we have one big operator called “read-out” or $\mathcal{R}$; this operator will measure all of the qubits in the system, and its value will tell us *exactly* what the state of the system is in terms of the computational basis.

i Two-Qubit Readout

The readout operator $\mathcal{R}$ maps the two-qubit basis states to unique numbers:

- $\mathcal{R}\,|00\rangle = 0\,|00\rangle$
- $\mathcal{R}\,|01\rangle = 1\,|01\rangle$
- $\mathcal{R}\,|10\rangle = 2\,|10\rangle$
- $\mathcal{R}\,|11\rangle = 3\,|11\rangle$

This binary-to-decimal conversion helps us uniquely identify each computational basis state.

However, this operator $\mathcal{R}$ *must* be Hermitian to be a physical observable, and as a result its eigenstates span our Hilbert space. Well, upon readout, we know that each qubit can be in one of two states $b_i = 0$ or 1, and if we have $i = 1, \dots, N$ qubits, there are 2^N possibilities.

We will return to this, but let us linger on two qubits. Notice that we can obtain the above states with an operation called the *tensor product*.

$\otimes$	$\|0\rangle$	$\|1\rangle$
$\|0\rangle$	$\|0\rangle \otimes \|0\rangle$	$\|0\rangle \otimes \|1\rangle$
$\|1\rangle$	$\|1\rangle \otimes \|0\rangle$	$\|1\rangle \otimes \|0\rangle$

This multiplication table is how we go from basis sets of single qubits to the basis set of two qubits. Very often, we will drop the $\otimes$ and simply write

$$|b_1 b_2\rangle \equiv |b_1\rangle \otimes |b_2\rangle \, .$$

Similarly, we can build up three qubit states by taking the tensor product of two-qubit states with a single qubit:

$\otimes$	$\|0\rangle$	$\|1\rangle$
$\|00\rangle$	$\|00\rangle \otimes \|0\rangle$	$\|00\rangle \otimes \|1\rangle$
$\|01\rangle$	$\|01\rangle \otimes \|0\rangle$	$\|01\rangle \otimes \|1\rangle$
$\|10\rangle$	$\|10\rangle \otimes \|0\rangle$	$\|10\rangle \otimes \|1\rangle$
$\|11\rangle$	$\|11\rangle \otimes \|0\rangle$	$\|11\rangle \otimes \|1\rangle$

again we can define

$$|b_1 b_2 b_3\rangle = |b_1 b_2\rangle \otimes |b_3\rangle \, ,$$

and we proceed once more

$\otimes$	$\|0\rangle$	$\|1\rangle$
$\|000\rangle$	$\|000\rangle \otimes \|0\rangle$	$\|000\rangle \otimes \|1\rangle$
$\|001\rangle$	$\|001\rangle \otimes \|0\rangle$	$\|001\rangle \otimes \|1\rangle$
$\|010\rangle$	$\|010\rangle \otimes \|0\rangle$	$\|010\rangle \otimes \|1\rangle$
$\|011\rangle$	$\|011\rangle \otimes \|0\rangle$	$\|011\rangle \otimes \|1\rangle$
$\|100\rangle$	$\|100\rangle \otimes \|0\rangle$	$\|100\rangle \otimes \|1\rangle$
$\|101\rangle$	$\|101\rangle \otimes \|0\rangle$	$\|101\rangle \otimes \|1\rangle$
$\|110\rangle$	$\|110\rangle \otimes \|0\rangle$	$\|110\rangle \otimes \|1\rangle$
$\|111\rangle$	$\|111\rangle \otimes \|0\rangle$	$\|111\rangle \otimes \|1\rangle$

As we can see, every time we add a new qubit, the dimension of the space is multiplied by two. In general, we can can break N-qubits in the computational basis as

$$|b_1 b_2 \cdots b_N\rangle = |b_1\rangle \otimes |b_2\rangle \otimes \cdots \otimes |b_N\rangle$$

where $b_j = 0$ or 1. Since this is a basis, we can "count" the number of basis states to determine the dimension of the space to obtain for the Hilbert space of N-qubits $\mathscr{H}_N$,

$$\dim \mathscr{H}_N = 2^N.$$

Therefore, a general quantum state of N qubits can be written as a superposition of all possible computational basis states:

$$|\psi\rangle = \sum_{b_1,\ldots,b_N=0,1} \psi_{b_1 \cdots b_N} |b_1 \cdots b_N\rangle$$

where $\psi_{b_1 \cdots b_N}$ are complex coefficients satisfying the normalization condition

$$\sum_{b_1,\ldots,b_N=0,1} |\psi_{b_1 \cdots b_N}|^2 = 1.$$

This means that to fully specify a quantum state of N qubits, we need 2^N complex numbers (subject to normalization), which illustrates both the power and challenge of quantum computing - the exponential growth in the state space allows for massive parallel processing but also makes classical simulation difficult.

Two-qubit wave functions and entanglement

A general two-qubit wave function can be written as

$$|\psi\rangle = \psi_{00}\,|00\rangle + \psi_{01}\,|01\rangle + \psi_{10}\,|10\rangle + \psi_{11}\,|11\rangle$$

where $|\psi_{00}|^2 + |\psi_{01}|^2 + |\psi_{10}|^2 + |\psi_{11}|^2 = 1$. When the two qubits are independent (a product state), we can write this as a tensor product of individual qubit states:

$$\begin{aligned} |\psi\rangle &= (\alpha_1\,|0\rangle + \beta_1\,|1\rangle) \otimes (\alpha_2\,|0\rangle + \beta_2\,|1\rangle) \\ &= \alpha_1\alpha_2\,|00\rangle + \alpha_1\beta_2\,|01\rangle + \beta_1\alpha_2\,|10\rangle + \beta_1\beta_2\,|11\rangle \end{aligned}$$

However, not all two-qubit states can be written as such a product! States that cannot be factored into a tensor product of individual qubit states are called *entangled states* - a crucial quantum resource we'll explore later.

The exponential growth in the state space has important implications for simulating quantum systems on classical computers. While we need 2^N complex numbers to specify an arbitrary quantum state, the situation becomes even more demanding when we consider operations on these states:

1. To represent an arbitrary quantum operation (unitary evolution) on N qubits, we need a $2^N \times 2^N$ unitary matrix. This requires storing and manipulating 2^{2N} complex numbers.
2. Even to compute the probability of a measurement outcome, we need to perform operations involving all 2^N amplitudes.

This exponential scaling is why classical computers struggle to simulate large quantum systems - the memory and computational requirements become overwhelming. For example:

- 10 qubits: $2^{10} = 1,024$ amplitudes, $2^{20} \approx 1$ million matrix elements
- 30 qubits: $2^{30} \approx 1$ billion amplitudes, $2^{60} \approx 10^{18}$ matrix elements
- 50 qubits: $2^{50} \approx 10^{15}$ amplitudes, $2^{100} \approx 10^{30}$ matrix elements

However, it's important to note that not all quantum computations require storing and manipulating the full state space. Many practical quantum algorithms and simulations exploit special properties:

- Some quantum states have special structure (like product states) that allow more efficient representations
- Many quantum operations act locally or have special symmetries that reduce the computational complexity
- Some quantum algorithms can be simulated using specialized techniques that avoid storing the full state vector

Nevertheless, the ability to access and manipulate this exponentially large state space can help us perform computations that classical computers would struggle with.

! Classical simulation vs. quantum measurement

When simulating quantum systems on classical computers, we have direct access to the full state vector - all the complex amplitudes $\psi_{b_1 \cdots b_N}$. This gives us complete information about the quantum state, allowing us to calculate any property without performing repeated measurements.
In contrast, real quantum computers are bound by the measurement postulates of quantum mechanics (Postulate II, Section 1.3.2). Each measurement:

1. Collapses the quantum state
2. Only returns eigenvalues of the measured observable
3. Must be repeated many times to estimate expectation values and state properties

This limitation of quantum hardware is why techniques like quantum state tomography are necessary - reconstructing the full quantum state requires performing many different measurements on multiple copies of the same state. Classical simulation sidesteps this fundamental quantum constraint, though at the cost of exponential classical resources.

Example: Building up three-qubit states

Consider how we build up the state $|101\rangle$:

1. Start with first two qubits: $|10\rangle$
2. Tensor with third qubit: $|10\rangle \otimes |1\rangle$
3. This gives: $|101\rangle$

We can verify this matches our counting:

- First qubit: $|1\rangle$ (second basis state)
- Second qubit: $|0\rangle$ (first basis state)
- Third qubit: $|1\rangle$ (second basis state)

Therefore in binary: 101, which is state number 5 in our computational basis (counting from 0).

Before we run head first into entanglement, let's take a minute to just do some counting to see that we are going to run into some trouble. For a single qubit, we have two complex numbers $|\psi\rangle = \alpha |0\rangle + \beta |1\rangle$, but we need to normalize them $|\alpha|^2 + |\beta|^2 = 1$ and remove a phase. Thus, we have reduced our 2 complex numbers (= 4 real numbers) down to 4-1-1=2 real numbers. This is made explicit with the Bloch sphere where two real numbers (θ, ϕ) completely characterize the state. Therefore, do we only need $2 \times 2 = 4$ real numbers to describe a 4 qubit state? Well, let's count, we can see from Eq. 3.1 that we have 4 complex numbers (= 8 real numbers). But we must also impose normalization and remove an overall phase, reducing us down to 8-1-1 = 6 real numbers. But $6 > 4$; this is our first hint of something happening in our quantum system, we need more numbers to describe all of the states in a two qubit system than simply what we needed for two separate qubits.

ℹ Mathematical Note: State Space Geometry

For the mathematically inclined: The physical state space of an N-qubit system is complex projective space $\mathbb{CP}^{2^N-1}$. For two qubits, this means $\mathbb{CP}^3$, which is fundamentally different from $\mathbb{CP}^1 \times \mathbb{CP}^1$ (the space of two separate qubits). This geometric fact underlies why we need more parameters to describe entangled states - the state space has a richer structure than just the product of individual qubit spaces.

3.2 Entanglement

Entanglement is one of the most important phenomena in quantum mechanics without a clear classical antecedent. The term was first coined by Schrödinger in 1935 [19] in response to a famous paper by Einstein, Podolsky, and Rosen (EPR) [20]. It represents quantum correlations between particles that cannot be explained by a simple "lack of knowledge" by the observer. To get around this, it was thought there must be "hidden variables" to make quantum mechanics complete; thus, EPR used these quantum correlations to argue that quantum mechanics must be incomplete. Quantum mechanics appeared to allow "spooky action at a distance" that violated their ideas of locality and reality.

However, in 1964, John Stewart Bell [21] showed that quantum mechanics predicts correlations between entangled particles that are mathematically impossible to explain with any local hidden variable theory. Subsequent experiments have repeatedly confirmed these "Bell inequality violations," demonstrating that entanglement represents a fundamentally new kind of physical relationship not reducible to classical correlations (early experiments include [22] and [23]).

The existence of entanglement suggests that the quantum wave function represents more than just our knowledge about measurement probabilities - it appears to be a real physical object. More recent work by Matthew Pusey, Jonathan Barrett, and Terry Rudolph [24] has strengthened this view through their "PBR theorem," which shows that if quantum predictions are correct, then quantum states must be physically real rather than merely statistical.

ℹ Reality of the Wave Function

The PBR theorem (2012) [24] tells us something deep about quantum mechanics. To quote the paper,

> In conclusion, we have presented a no-go theorem, which—modulo assumptions—shows that models in which the quantum state is interpreted as mere information about an objective physical state of a system cannot reproduce the predictions of quantum theory. The result is in the same spirit as Bell's theorem, which states that no local theory can reproduce the predictions of quantum theory.

This provides strong support for viewing entanglement as a genuine physical phenomenon rather than just a limitation of our knowledge.

Let's explore what entanglement means mathematically and physically.

3.2.1 Bell States and Non-local Correlations

One of the simplest cases of quantum entanglement is the Bell state

$$|\Phi^+\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}} \tag{3.2}$$

This state exhibits perfect correlations that persist regardless of the physical separation between the qubits. When we measure the first qubit:

- If we get $|0\rangle$, the state collapses to $|00\rangle$
- If we get $|1\rangle$, the state collapses to $|11\rangle$

The remarkable feature is that measuring either qubit instantly determines the state of the other qubit, even if they are separated by vast distances. For example:

1. Create $|\Phi^+\rangle$ and separate the qubits by sending one to Earth and one to Mars
2. Measure the Earth qubit → get result 0 or 1 with 50% probability
3. We know with 100% certainty that the Mars qubit will be measured to give the same result.
4. This happens faster than light could travel between the qubits.

❗ No faster-than-light communication

While entanglement appears to create "spooky action at a distance," it cannot be used to transmit information faster than light. This is because:

1. The measurement results are random
2. The person with the second qubit needs classical information about the first measurement to interpret their results
3. This classical information is still limited by the speed of light

ℹ Bell state properties are basis independence

A remarkable property of the Bell state $|\Phi^+\rangle$ is that these perfect correlations persist no matter what basis we measure in. If we measure the first qubit in any basis and get some state $|\psi\rangle$, the second qubit will always be found in state $|\psi\rangle$ when measured in the same basis.
For example, if we measure the first qubit in the X basis: - If we get $|+\rangle$, the state collapses to $|{++}\rangle$ - If we get $|-\rangle$, the state collapses to $|{--}\rangle$
This is because we can rewrite $|\Phi^+\rangle$ in any basis and it maintains the same form:

$$|\Phi^+\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}} = \frac{|{++}\rangle + |{--}\rangle}{\sqrt{2}}$$

This is true for any basis, not just Z and X.

These non-local correlations are fundamentally different from classical correlations, but how can we see that? The key ends up being: measurements that show quantum correlations and was Bell's central insight [21]. Instead of just looking at measurements of Z which are easily explained by a classical hidden variable, also perform measurements at other angles of the Bloch sphere.

ℹ Mermin's Example

To see a simple example, read [25]. The essence of the idea presented there is to measure three different observables with the Bell state in Eq. 3.2,

$$Z, \quad -\frac{1}{2}Z + \frac{\sqrt{3}}{2}X, \quad -\frac{1}{2}Z - \frac{\sqrt{3}}{2}X,$$

with the results from these three measurements, there is no way to use classical probabilities of measurement outcomes to account for the distribution of results.

3.2.2 Mathematical definition and separability

A multi-qubit quantum state is entangled *if and only if* it cannot be written as a tensor product of individual qubit states[1]. For a two-qubit pure state $|\psi\rangle$, this means there do not exist single-qubit states $|\phi_1\rangle$ and $|\phi_2\rangle$ such that:

$$|\psi\rangle = |\phi_1\rangle \otimes |\phi_2\rangle$$

For a general two-qubit state $|\psi\rangle = \alpha|00\rangle + \beta|01\rangle + \gamma|10\rangle + \delta|11\rangle$, there is a simple condition for separability: the state is separable if and only if the determinant of its coefficient matrix is zero:

$$\begin{vmatrix} \alpha & \beta \\ \gamma & \delta \end{vmatrix} = \alpha\delta - \beta\gamma = 0$$

This can be proven by writing out the general form of a tensor product and matching coefficients. The classic examples of maximally entangled states are the Bell states:

$$|\Phi^+\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}} \qquad |\Phi^-\rangle = \frac{|00\rangle - |11\rangle}{\sqrt{2}}$$
$$|\Psi^+\rangle = \frac{|01\rangle + |10\rangle}{\sqrt{2}} \qquad |\Psi^-\rangle = \frac{|01\rangle - |10\rangle}{\sqrt{2}}$$

For mixed states, the situation is more complex and requires measures like concurrence to fully characterize entanglement.

Example: Separable state condition

Consider a separable two-qubit state formed by the tensor product of two arbitrary single-qubit states:

$$|\phi_1\rangle = a|0\rangle + b|1\rangle \qquad |\phi_2\rangle = c|0\rangle + d|1\rangle$$

Their tensor product gives:

[1] This is true for pure states, but can be easily generalized to mixed states, see [7].

$$\begin{aligned}|\phi_1\rangle \otimes |\phi_2\rangle &= (a|0\rangle + b|1\rangle) \otimes (c|0\rangle + d|1\rangle) \\ &= ac|00\rangle + ad|01\rangle + bc|10\rangle + bd|11\rangle\end{aligned}$$

The coefficient matrix determinant is:

$$\begin{vmatrix} ac & ad \\ bc & bd \end{vmatrix} = (ac)(bd) - (ad)(bc) = abcd - abcd = 0$$

This confirms that any separable state satisfies the zero determinant condition. Conversely, if a state's coefficient matrix has non-zero determinant, it must be entangled.

Example: Checking for Entanglement

Let's examine two states to see if they're entangled:

1. Consider the state $|\psi_1\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ The coefficient matrix is:

$$\begin{pmatrix} 1/\sqrt{2} & 0 \\ 0 & 1/\sqrt{2} \end{pmatrix}$$

The determinant is $(1/\sqrt{2})(1/\sqrt{2}) - (0)(0) = 1/2 \neq 0$, so this state is entangled.

2. Consider the state $|\psi_2\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |01\rangle)$ The coefficient matrix is:

$$\begin{pmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 0 & 0 \end{pmatrix}$$

The determinant is $(1/\sqrt{2})(0) - (1/\sqrt{2})(0) = 0$, so this state is separable. Indeed, we can write it as $|\psi_2\rangle = |0\rangle \otimes \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$.

3. Consider the state $|\psi_3\rangle = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)$ The coefficient matrix is:

$$\begin{pmatrix} 1/2 & 1/2 \\ 1/2 & 1/2 \end{pmatrix}$$

The determinant is $(1/2)(1/2) - (1/2)(1/2) = 0$, so this state is separable. We can verify this by rewriting it as: $|\psi_3\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$

3.2.3 Reduced density matrices

For entangled states, we often want to describe the state of a single qubit within the two-qubit system. This is done using the reduced density matrix, obtained by "tracing out" the other qubit. The process, called the partial trace, gives us a density matrix that describes all measurable properties of the subsystem we're interested in.

For a two-qubit system in state $|\psi\rangle$, the density matrix is $\rho = |\psi\rangle\langle\psi|$. To get the reduced density matrix for the first qubit (ρ_A), we take the partial trace over the second qubit (B):

$$\rho_A = \mathrm{Tr}_B(\rho) = \sum_{b=0}^{1} \langle b_B|\rho|b_B\rangle$$

For a single term $\langle b_B|\rho|b_B\rangle$, this gives a 2×2 matrix acting on the first qubit:

$$\begin{aligned}\langle b_B|\rho|b_B\rangle &= \begin{pmatrix} \langle 0b|\rho|0b\rangle & \langle 0b|\rho|1b\rangle \\ \langle 1b|\rho|0b\rangle & \langle 1b|\rho|1b\rangle \end{pmatrix} \\ &= \begin{pmatrix} |\langle 0b|\psi\rangle|^2 & \langle 0b|\psi\rangle \langle 1b|\psi\rangle^* \\ \langle 1b|\psi\rangle \langle 0b|\psi\rangle^* & |\langle 1b|\psi\rangle|^2 \end{pmatrix}\end{aligned}$$

where $|b_B\rangle$ are the basis states of qubit B. Each element of ρ_A is a sum of two elements from the original density matrix, corresponding to tracing out qubit B in the computational basis.

Tensor notation for density matrices

In tensor notation, a two-qubit state $|\psi\rangle$ has components ψ_{ij} where i, j label the basis states of the first and second qubit. The density matrix elements are then $\rho_{ij,kl} = \psi_{ij}\psi^*_{kl}$, where the first pair of indices (i, j) corresponds to the ket and (k, l) to the bra.
The partial trace over qubit B corresponds to summing over matching indices for qubit B:

$$(\rho_A)_{ik} = \sum_j \rho_{ij,kj}$$

This tensor notation makes it clear why this operation is called a "trace" - we're summing over diagonal elements where the indices for system B match (j=j), just like in the usual matrix trace, while keeping the indices for system A (i,k) free.

Example: Reduced density matrix of a Bell state

Consider the Bell state $|\Phi^+\rangle = \frac{|00\rangle+|11\rangle}{\sqrt{2}}$. Its density matrix is:

$$\rho = |\Phi^+\rangle\langle\Phi^+| = \frac{1}{2}(|00\rangle\langle 00| + |00\rangle\langle 11| + |11\rangle\langle 00| + |11\rangle\langle 11|)$$

Let's find ρ_A step by step:

1. First, we compute the partial trace:

$$\begin{aligned}\rho_A &= \langle 0_B|\rho|0_B\rangle + \langle 1_B|\rho|1_B\rangle \\ &= \frac{1}{2}|0\rangle\langle 0| + \frac{1}{2}|1\rangle\langle 1| \\ &= \frac{1}{2}I\end{aligned}$$

2. This shows that when we look at just one qubit of a maximally entangled pair:
   - It appears to be in a completely mixed state
   - We have equal probability of measuring 0 or 1
   - All quantum information is stored in the correlations between qubits

The reduced density matrix reveals a key feature of entanglement: while the total state is pure ($\rho^2 = \rho$), the subsystem state can be mixed ($\rho_A^2 \neq \rho_A$). This is a signature of entanglement - if we can only access one qubit of an entangled pair, we see a statistical mixture rather than a pure state.

Example: Partial Trace for a General Two-Qubit State

Suppose we have a general two-qubit pure state:

$$|\psi\rangle = \alpha|00\rangle + \beta|01\rangle + \gamma|10\rangle + \delta|11\rangle.$$

The full density matrix is:

$$\rho = |\psi\rangle\langle\psi| = \begin{pmatrix} \alpha \\ \beta \\ \gamma \\ \delta \end{pmatrix} \begin{pmatrix} \alpha^* & \beta^* & \gamma^* & \delta^* \end{pmatrix}.$$

To find the reduced density matrix of the first qubit, $\rho_A = \mathrm{Tr}_B(\rho)$, group the basis states so that qubit B's index is traced out:

$$\rho_A = \begin{pmatrix} |\alpha|^2 + |\beta|^2 & \alpha\gamma^* + \beta\delta^* \\ \gamma\alpha^* + \delta\beta^* & |\gamma|^2 + |\delta|^2 \end{pmatrix}.$$

This 2×2 matrix captures all local measurements and observables on qubit A, regardless of the state of qubit B.

i Why Care About Reduced Density Matrices?

Reduced density matrices are crucial because they tell us what we can observe when we only have access to part of an entangled system. They help answer questions like:

1. **Local Measurements**: What results will we get if we only measure one qubit of an entangled pair?
2. **Quantum Information**: How much information is accessible locally vs. stored in correlations?
3. **Decoherence**: How does interaction with the environment affect our quantum system?

For example, in quantum teleportation, while the total state remains pure, the reduced density matrix of the transmitted qubit appears completely mixed until the classical information is received. This explains why teleportation cannot transmit information faster than light!

3.2.4 Quantifying entanglement

Several measures exist to quantify entanglement, each capturing different aspects:

1. **Von Neumann entropy**: For a pure bipartite state $|\psi\rangle$, the entanglement entropy is $S(\rho_A) = -\mathrm{Tr}(\rho_A \log_2 \rho_A)$ where ρ_A is the reduced density matrix of subsystem A. For two qubits, this ranges from 0 for separable states to 1 for maximally entangled states.

2. **Concurrence** [26]: For a two-qubit state ρ, defined as $C(\rho) = \max(0, \lambda_1 - \lambda_2 - \lambda_3 - \lambda_4)$ where λ_i are the square roots of eigenvalues of $\rho(Y \otimes Y)\rho^*(Y \otimes Y)$ in decreasing order (this reduces to $2|\alpha\delta - \beta\gamma|$ for a pure state $|\psi\rangle = \alpha|00\rangle + \beta|01\rangle + \gamma|10\rangle + \delta|11\rangle$).
3. **Negativity**: Based on the partial transpose of the density matrix, providing a computable measure that captures the degree of entanglement. However, while negativity is zero for separable states, a zero negativity does not guarantee separability for some higher-dimensional systems - there exist entangled states with zero negativity.

These measures help quantify the "quantum-ness" of correlations and their potential utility in quantum information protocols. All of these are discussed in detail in [7].

We will find that many protocols of usefulness will produce entanglement in the system, though often only when in a particular basis. We will see that in the next section when we introduce operators on this Hilbert space

! Entanglement depends on the partition

When we talk about entanglement between subsystems A and B, it's crucial to understand that this depends entirely on how we choose to divide our total system into these subsystems. The same quantum state can appear entangled or unentangled depending on this choice of partition. For example, consider the state:

$$|\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle),$$

this is clearly an entangled Bell state, but our Hilbert space exists as (complex) four-dimensional space. I could "relabel" my states into a new basis (which we can define with a tilde), such that

$$\begin{aligned}
|00\rangle &= \frac{1}{\sqrt{2}}(|\tilde{0}\tilde{0}\rangle + |\tilde{1}\tilde{1}\rangle), \\
|11\rangle &= \frac{1}{\sqrt{2}}(|\tilde{0}\tilde{0}\rangle - |\tilde{1}\tilde{1}\rangle), \\
|01\rangle &= \frac{1}{\sqrt{2}}(|\tilde{0}\tilde{1}\rangle + |\tilde{1}\tilde{0}\rangle), \\
|10\rangle &= \frac{1}{\sqrt{2}}(|\tilde{0}\tilde{1}\rangle - |\tilde{1}\tilde{0}\rangle),
\end{aligned}$$

and if we do that, we see that

$$|\psi\rangle = |\tilde{0}\tilde{0}\rangle\,,$$

a completely disentangled state! This is not to say that $|\psi\rangle$ is not entangled, it is. It is merely important to remember that entanglement is defined with respect to a certain partitioning of your Hilbert space. The space represented by $\tilde{0}$ and $\tilde{1}$ is a complex combination of 00, 01, 10, and 11. It is usually the physical situation which dictates how we partition our system (systems that are physically isolated from each other or for which we have single degrees of freedom which admit simple tensor products when considered with other systems).

When to Use Different Entanglement Measures

Each entanglement measure has its strengths:

1. **Von Neumann Entropy**
   - Best for: Pure bipartite states
   - Advantages: Clear physical interpretation, easy to calculate
   - Use when: You want to quantify how much quantum information is shared between subsystems
2. **Concurrence**
   - Best for: Two-qubit mixed states
   - Advantages: Can be directly calculated from density matrix
   - Use when: Working with noisy or mixed two-qubit states
3. **Negativity**
   - Best for: Higher-dimensional systems
   - Advantages: Easy to compute, works for mixed states
   - Use when: Dealing with larger systems or when you need a quick estimate of entanglement

Example: For the Bell state $|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$:

- Von Neumann Entropy = 1 (maximally entangled)
- Concurrence = 1 (maximally entangled)
- Negativity = 0.5 (maximum value for two qubits)

Three qubits can be entangled in two inequivalent ways @Dur2000

Three-qubit entanglement introduces fundamentally new features not present in two-qubit systems. The most famous example is the GHZ state (named after Greenberger, Horne, and Zeilinger [27]):

$$|\mathrm{GHZ}\rangle = \frac{1}{\sqrt{2}}(|000\rangle + |111\rangle)$$

Unlike two-qubit entanglement, which has only one type of maximal entanglement (equivalent to Bell states), three-qubit systems can exhibit qualitatively different kinds of entanglement. The GHZ state above has the special property that measuring any one qubit immediately determines the state of the other two, but if you lose (trace out) any one qubit, the remaining two qubits are completely unentangled. This is fundamentally different from another type of three-qubit entanglement called the W state:

$$|W\rangle = \frac{1}{\sqrt{3}}(|001\rangle + |010\rangle + |100\rangle)$$

which maintains some two-qubit entanglement even after losing one qubit. These distinct classes of entanglement cannot be converted into each other using local operations and classical communication (LOCC).

3.3 Multi-qubit operations

With our exponentially big state space created and entanglement characterized, we can begin to think about how to translate our single qubit operations to multiple qubits and then how to build up operations that *use* multiple qubits. We will focus on two qubit gates since these will help us build up a set of universal gates for quantum computation.

3.3.1 Single-qubit gates

When we want to apply a single-qubit gate to one qubit in a multi-qubit system, we need to use tensor products to construct the appropriate operator. For a two-qubit system, if we want to apply a gate U to the first qubit, the full operator is:

$$U \otimes I$$

where I is the 2×2 identity matrix. Similarly, to apply U to the second qubit, we use:

$$I \otimes U$$

For example, applying the Pauli-X gate to the first qubit of a two-qubit system gives:

$$X \otimes I = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \otimes \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

where the rows and columns correspond to the basis states in order $|00\rangle$, $|01\rangle$, $|10\rangle$, $|11\rangle$. For example, the 1 in the third row, first column means $X \otimes I$ transforms $|00\rangle$ to $|10\rangle$, which flips the first qubit as expected.

Similarly, applying it to the second qubit gives:

$$I \otimes X = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \otimes \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Here, the 1 in the second row, first column shows $I \otimes X$ transforms $|00\rangle$ to $|01\rangle$, flipping only the second qubit as expected.

This pattern extends to more qubits. For an N-qubit system, to apply U to the kth qubit, we use:

$$\underbrace{I \otimes \cdots \otimes I}_{k-1} \otimes U \otimes \underbrace{I \otimes \cdots \otimes I}_{N-k}$$

where U appears in the kth position and I appears in all other positions. This construction ensures we affect only the target qubit while leaving all other qubits unchanged.

Example: H gate on second qubit of three-qubit system

Let's see how applying H to the second qubit of a three-qubit system works. The operator is:

$$I \otimes H \otimes I$$

Acting on the state $|000\rangle$:

$$\begin{aligned}(I \otimes H \otimes I)|000\rangle &= (I \otimes H \otimes I)(|0\rangle \otimes |0\rangle \otimes |0\rangle) \\ &= (I|0\rangle) \otimes (H|0\rangle) \otimes (I|0\rangle) \\ &= |0\rangle \otimes \frac{|0\rangle + |1\rangle}{\sqrt{2}} \otimes |0\rangle \\ &= \frac{|000\rangle + |010\rangle}{\sqrt{2}}\end{aligned}$$

As expected, only the middle qubit is put into an equal superposition of 0 and 1.

! Notational convenience for single-qubit operations

To avoid writing long chains of tensor products, we often use a subscript to indicate which qubit an operator acts on. For example, instead of writing

$$I \otimes X \otimes I \otimes I \otimes X$$

for a five-qubit system where we apply X to qubits 2 and 5, we can write this more compactly as

$$X_2 X_5$$

Similarly, applying the Hadamard gate to the third qubit of a four-qubit system would be written as

$$H_3$$

rather than $I \otimes I \otimes H \otimes I$. This notation is particularly helpful when describing quantum circuits involving many qubits.

3.3.2 Two-qubit gates

To begin to build up a full set of logical gates, we start with "controlled" gates. These are gates that will act as single qubit gates on one qubit (the target), but only if another qubit (the control) is in the $|1\rangle$ state.

We first introduce the two-qubit gate called the controlled-NOT (CNOT) gate, which flips the second qubit (target) when the first qubit (control) is in state $|1\rangle$[2].

The CNOT gate can be written as a 4×4 matrix acting on the two-qubit basis states $|00\rangle$, $|01\rangle$, $|10\rangle$, and $|11\rangle$:

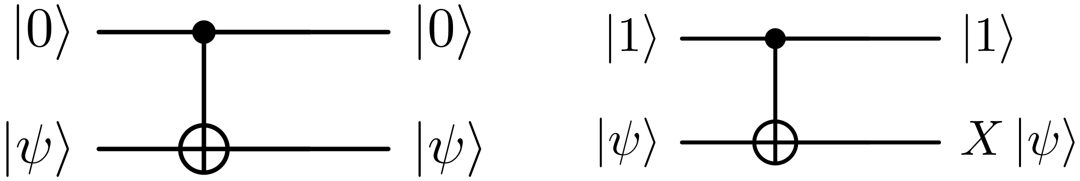
[2] This is also known as the CX-gate (the controlled-X gate).

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

We can understand this matrix by seeing how it acts on each basis state:

$$\begin{aligned} \text{CNOT}|00\rangle &= |00\rangle \\ \text{CNOT}|01\rangle &= |01\rangle \\ \text{CNOT}|10\rangle &= |11\rangle \\ \text{CNOT}|11\rangle &= |10\rangle \end{aligned}$$

When the first (control) qubit is $|0\rangle$, the target qubit is unchanged. When the control qubit is $|1\rangle$, the target qubit is flipped (X gate applied). This has a circuit representation which we show below in Fig. 3.1.



(a) CNOT gate with control in $|0\rangle$

(b) CNOT gate with control in $|1\rangle$

Fig. 3.1: The CNOT gate behavior depends on the control qubit state. When the control is $|0\rangle$ (left), the target is unchanged. When the control is $|1\rangle$ (right), the target is flipped.

The CNOT gate's action on computational basis states appears simple - it either leaves them unchanged (when control is |0) or flips the target (when control is |1). However, when applied to superposition states, the CNOT can create entanglement.

For example, consider applying CNOT to a superposition state created by applying a Hadamard gate to the first qubit (circuit diagram in Fig. 3.2).

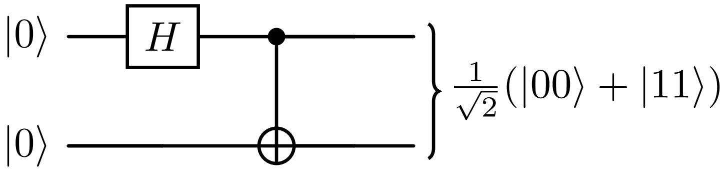


Fig. 3.2: The application of H followed by a CNOT gate can create the Bell state $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$.

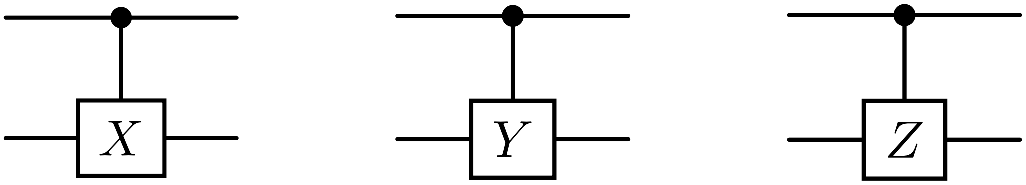
We can verify this mathematically:

$$\begin{aligned} |00\rangle &\xrightarrow{H\otimes I} \frac{|0\rangle + |1\rangle}{\sqrt{2}} \otimes |0\rangle = \frac{|00\rangle + |10\rangle}{\sqrt{2}} \\ &\xrightarrow{\text{CNOT}} \frac{|00\rangle + |11\rangle}{\sqrt{2}} \end{aligned}$$

The resulting state is the Bell state $|\Phi^+\rangle$ which we saw earlier - a maximally entangled state that cannot be written as a product of individual qubit states.

The CNOT gate is just one example of a controlled operation. More generally, we can create controlled versions of any single-qubit gate, where the target operation is applied only when the control qubit is $|1\rangle$. The most common controlled gates are:

- Controlled-X (CNOT): Flips the target qubit if control is $|1\rangle$, Fig. 3.3a
- Controlled-Z (CZ): Adds -1 phase if both qubits are $|1\rangle$, Fig. 3.3c
- Controlled-Y (CY): Applies Y rotation if control is $|1\rangle$, Fig. 3.3b



(a) CX

(b) CY

(c) CZ

Fig. 3.3: Circuit Notation for controlled-X (CX), controlled-Y (CY), and controlled-Z (CZ) gates. Note that CX=CNOT, so this is just an alternative circuit notation.

Any controlled gate can be constructed using CNOT gates and single-qubit operations. For example, the CZ gate can be implemented as:

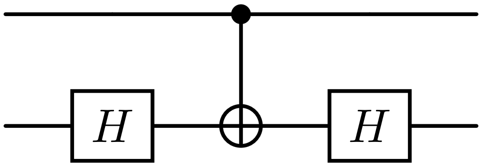


Fig. 3.4: The CZ gate can be decomposed into two Hadamards sandwiching a CNOT (CX) gate.

This notation is flexible as well, we can begin to apply any unitary matrix U onto a target qubit and have a control qubit for it. This generically will look like

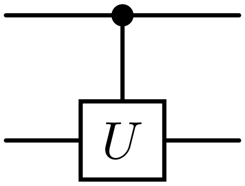


Fig. 3.5: An arbitrary controlled gate

Mathematically, we can write this out in a four-by-four matrix

$$\mathrm{CU} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & U_{11} & U_{12} \\ 0 & 0 & U_{21} & U_{22} \end{pmatrix}$$

where U_{ij} are the matrix elements of the single-qubit unitary $U = \begin{pmatrix} U_{11} & U_{12} \\ U_{21} & U_{22} \end{pmatrix}$.

Another important two-qubit gate is the SWAP gate, which exchanges the states of two qubits. The SWAP gate can be written as a four-by-four matrix:

$$\text{SWAP} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Unlike controlled gates, the SWAP gate cannot create entanglement - it simply swaps the quantum states between the qubits. The circuit notation for a SWAP gate is:

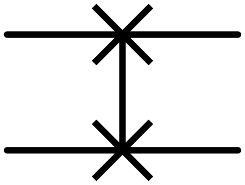


Fig. 3.6: SWAP gate circuit notation

We've seen how this can already create entanglement even though one qubit is merely acting as "control," and how we can "SWAP" two qubits. However, we can also write down more general four-by-four matrices on two qubits and if we have a generic four-by-four matrix U_4, we will write that circuit element as

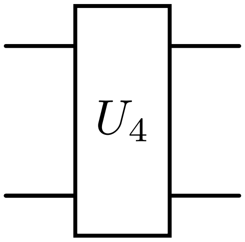


Fig. 3.7: A generic unitary on two qubits can be represented by a block spanning the two qubits

These two-qubit gates form the foundation for quantum computation with multiple qubits. The controlled operations, particularly the CNOT gate, are essential building blocks for quantum algorithms, while general two-qubit unitaries give us the full power to manipulate quantum states in ways impossible with just single-qubit operations. Understanding how these gates create and manipulate entanglement is crucial.

Example: Creating different entangled states

Let's see how different combinations of gates create distinct entangled states:

1. Bell state $|\Phi^+\rangle$:

```
|0  --H--•--|
        |
|0 -----X--|
```

$$|00\rangle \xrightarrow{H\otimes I} \frac{|00\rangle+|10\rangle}{\sqrt{2}} \xrightarrow{CNOT} \frac{|00\rangle+|11\rangle}{\sqrt{2}}$$

2. Bell state $|\Psi^+\rangle$:

```
|0 --H--•--|
        |
|0 --X--X--|
```

$$|00\rangle \xrightarrow{I\otimes X} |01\rangle \xrightarrow{H\otimes I} \frac{|01\rangle+|11\rangle}{\sqrt{2}} \xrightarrow{CNOT} \frac{|01\rangle+|10\rangle}{\sqrt{2}}$$

This shows how different gate sequences can create different types of entanglement. *Note: We have used the fixed-width font notation for circuit diagrams here.*

3.3.2.1 Universal gate sets

Just as classical computation can be performed using a small set of universal gates (like NAND or NOR), quantum computation can be achieved using a finite set of quantum gates that can approximate any unitary operation to arbitrary precision. This is known as a universal gate set.

A common universal gate set consists of:

1. The CNOT gate
2. Single-qubit rotations (or equivalently, any set of gates that can approximate any single-qubit rotation)

Remarkably, these are sufficient to construct any unitary operation on any number of qubits, though the construction may require many gates. This is analogous to how NAND gates can be used to build any classical logic circuit.

There are several alternative universal gate sets. Some common ones include:

- CZ (or CY) + single-qubit rotations
- CNOT + Hadamard + Phase (S) gate + T gate
- Toffoli + Hadamard

The choice of which universal gate set to use often depends on the physical implementation of the quantum computer. For example, some quantum computing architectures might naturally implement CZ gates rather than CNOT gates, making the CZ-based universal set more practical.

It's worth noting that while we can approximate any unitary to arbitrary precision with these gate sets, the number of gates required might grow exponentially with the desired precision. This is known as the Solovay-Kitaev theorem, which provides an algorithm for finding such approximations.

Example: Creating a GHZ State

A three-qubit GHZ state is

$$|\text{GHZ}\rangle = \frac{|000\rangle + |111\rangle}{\sqrt{2}}.$$

We can create this state using the following sequence of gates:

1. Start from $|000\rangle$
2. Apply a Hadamard gate to the first qubit:

$$|000\rangle \xrightarrow{H \otimes I \otimes I} \frac{|000\rangle + |100\rangle}{\sqrt{2}}$$

3. Apply two consecutive CNOTs, using the first qubit as the control:

$$\frac{|000\rangle + |100\rangle}{\sqrt{2}} \xrightarrow{\text{CNOT on qubits } 1 \to 2} \frac{|000\rangle + |110\rangle}{\sqrt{2}}$$
$$\xrightarrow{\text{CNOT on qubits } 1 \to 3} \frac{|000\rangle + |111\rangle}{\sqrt{2}}$$

Measuring any qubit in the computational basis will instantly project the entire system into either $|000\rangle$ or $|111\rangle$, illustrating the strong correlations present in multipartite entanglement.

3.3.3 Measurement

When measuring multiple qubits, we need to extend our understanding of single-qubit measurements. For a single qubit, measuring in the computational basis was straightforward - we would get either $|0\rangle$ or $|1\rangle$. However, in a multi-qubit system, measuring just one qubit introduces an important concept: partial measurements and degeneracy.

Consider measuring the first qubit of a two-qubit state in the computational basis. The measurement operators are:

$$P_0 = |0\rangle\langle 0| \otimes I = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$P_1 = |1\rangle\langle 1| \otimes I = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

These operators are degenerate - for example, P_0 projects onto both $|00\rangle$ and $|01\rangle$ states. This means when we measure the first qubit and get 0, the second qubit remains in a quantum state.

Example: Partial Measurement

Consider the state:

$$|\psi\rangle = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)$$

If we measure the first qubit and get 0, the state collapses to:

$$|\psi'\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |01\rangle)$$

The second qubit remains in a superposition!

When measuring multiple qubits, we need to be careful about how we calculate probabilities. For a single qubit, the probability of measuring $|0\rangle$ was simply $|\langle 0|\psi\rangle|^2$. However, with multiple qubits, we need to sum over all possible configurations of the unmeasured qubits.

For example, if we have a two-qubit state $|\psi\rangle$ and measure only the first qubit, the probability of getting 0 is:

$$p(0) = \sum_i |\langle 0i|\psi\rangle|^2$$

where i runs over all possible states of the second qubit (0 and 1). This sum accounts for all ways we could get outcome 0 on the first qubit, regardless of what state the second qubit is in.

Example: Calculating Measurement Probabilities

Consider again the state:

$$|\psi\rangle = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)$$

The probability of measuring 0 on the first qubit is:

$$\begin{aligned} p(0) &= |\langle 00|\psi\rangle|^2 + |\langle 01|\psi\rangle|^2 \\ &= \left|\frac{1}{2}\right|^2 + \left|\frac{1}{2}\right|^2 \\ &= \frac{1}{2} \end{aligned}$$

This matches our intuition - the first qubit is equally likely to be 0 or 1.

3.3.3.1 Measuring General Observables

In practice, we may be restricted in what we can measure on a quantum computer, such as only in the computational (Z) basis. However, we often need to measure other observables. For example, we might want to measure the parity of two qubits ($Z_1 Z_2$) or their correlation ($X_1 X_2$).

To measure these observables, we need to transform our state before measurement. This is done by applying appropriate unitary operations that map our desired measurement basis to the computational basis.

For example, as we saw in the single qubit section, to measure in the X basis, we first apply H before measuring in the computational basis.

For two-qubit observables like $Z_1 Z_2$, we can use controlled operations to map the eigenspaces to computational basis states. Here's how we might measure $Z_1 Z_2$.

First, let's write out how $Z_1 Z_2$ acts on the computational basis, separting out the +1 eigenvalues from the -1 eigenvalues

State	$Z_1 Z_2$	Z_2
$\lvert 00\rangle$	$+1$	$+1$
$\lvert 01\rangle$	-1	-1
$\lvert 10\rangle$	-1	$+1$
$\lvert 11\rangle$	$+1$	-1

Notice that these operators have the same number of +1 and -1 eigenvalues. On top of that, the states $|00\rangle$ and $|01\rangle$ have the same eigenvalue, and $|10\rangle$ and $|11\rangle$ flips. This sounds like a CNOT gate, in fact if we have a unitary that takes $|10\rangle \mapsto |11\rangle$ and $|11\rangle \mapsto |10\rangle$, while leaving $|00\rangle$ and $|01\rangle$ alone, these operators match.

Let's build out a way to relate these operators which we will then use to measure *any* string Pauli matrices. With the above, we note that

$$\begin{aligned} |00\rangle &\mapsto |00\rangle\,, \\ |01\rangle &\mapsto |01\rangle\,, \\ |10\rangle &\mapsto |11\rangle\,, \\ |11\rangle &\mapsto |10\rangle \end{aligned}$$

will let us map $Z_1 Z_2 \mapsto Z_2$. This has a simple matrix form

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}.$$

In other words, the CNOT gate. Formally, this means that

$$\text{CNOT}\, Z_1 Z_2\, \text{CNOT} = Z_2, \tag{3.3}$$

which can be verified with matrix mulitplation. However, we can also note that if we apply $|b_1 b_2\rangle$ to both sides,

$$Z_1 Z_2 \text{CNOT}\,|b_1 b_2\rangle = \text{CNOT}(-1)^{b_2}\,|b_1 b_2\rangle = (-1)^{b_2}\text{CNOT}\,|b_1 b_2\rangle\,,$$

which verifies that CNOT $|b_1 b_2\rangle$ is an eigenvector of $Z_1 Z_2$ with eigenvalue $(-1)^{b_2}$ (this assume Eq. 3.3 is correct).

Therefore, to *measure* $Z_1 Z_2$, we perform CNOT, then measure the target qubit (number two). The circuit diagram looks like this

and if we want to not only measure $Z_1 Z_2$ but ensure the system goes into an eigenstate of $Z_1 Z_2$, we need to perform CNOT again

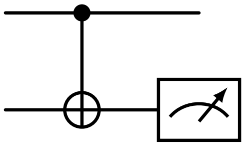
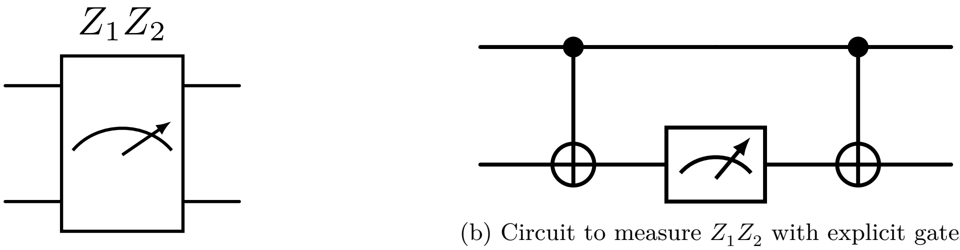


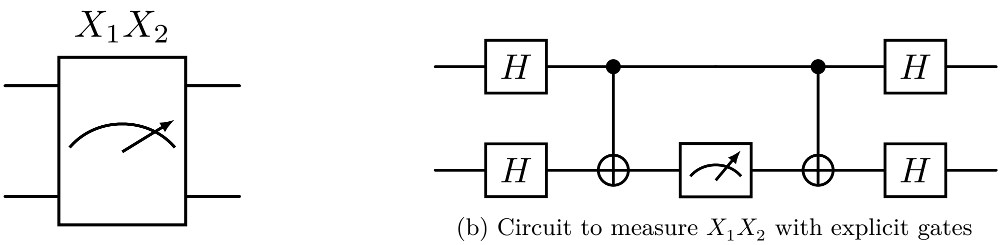
Fig. 3.8: Circuit to measure $Z_1 Z_2$



(b) Circuit to measure $Z_1 Z_2$ with explicit gates

(a) Circuit to measure $Z_1 Z_2$ with a single symbol

Fig. 3.9: Circuit diagrams for the $Z_1 Z_2$ measurement. These two circuits are equivalent to each other.



(b) Circuit to measure $X_1 X_2$ with explicit gates

(a) Circuit to measure $X_1 X_2$ with a single symbol

Fig. 3.10: Circuit diagrams for the $X_1 X_2$ measurement. These two circuits are equivalent to each other.

But what about measuring other Pauli strings? We can use the fact that $X = HZH$ and $Y = SXS^\dagger$ to convert any Pauli string measurement into a Z-type measurement. For example, to measure X_1X_2, we can apply Hadamard gates to both qubits, measure Z_1Z_2, and then apply Hadamard gates again:

This gives us a way to measure any Pauli string

1. Perform single-particle unitaries to convert it to only Z and I operators.
2. Find the permutation matrix that relates +1 and -1 eigenvalues of your Z-Pauli string to a single Z_n and apply that unitary matrix (this could be a complicated combination of CNOT gates)
3. Measure Z_n.
4. If the correct measured state is needed, undo the unitary in #2 followed by the single-particle unitaries in #1.

Pauli strings are particularly "simple." Measuring more complicated observables requires more thought and sometimes ancillary systems to assist in that measurement.

Example: Measuring X on One Qubit of a Two-Qubit System

Suppose we have a two-qubit state:

$$|\phi\rangle = \alpha|00\rangle + \beta|01\rangle + \gamma|10\rangle + \delta|11\rangle,$$

and we want to measure the first qubit in the X basis while leaving the second qubit unmeasured.

1. Recall $HXH = Z$. So to measure X on the first qubit, apply a Hadamard on that qubit to map the X basis to the Z basis:
$$(H \otimes I)|\phi\rangle.$$
2. Measure the first qubit in the computational basis (effectively measuring Z on the transformed state).
3. After the measurement, unapply the H if you want to restore the original basis of the first qubit.

This procedure effectively measures X_1 while leaving qubit 2 untouched (can it still be entangled?).

Example: Creating entanglement through measurement

Let's see how we can create entanglement through measurement. We'll start with the product state $|{+}{+}\rangle$ and use a Z_1Z_2 measurement to create a maximally entangled state.

1. Initial state:
$$|{+}{+}\rangle = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)$$
2. Measuring Z_1Z_2 projects onto the +1 or -1 eigenspaces:
   - +1 eigenspace: span$\{|00\rangle, |11\rangle\}$
   - -1 eigenspace: span$\{|01\rangle, |10\rangle\}$
3. After measurement:

- If outcome = +1:
$$|\Phi^+\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}}$$
- If outcome = -1:
$$|\Psi^+\rangle = \frac{|01\rangle + |10\rangle}{\sqrt{2}}$$

4. If we get -1, we can apply X to the second qubit to transform to $|\Psi^+\rangle$:

$$(I \otimes X)\frac{|01\rangle + |10\rangle}{\sqrt{2}} = \frac{|00\rangle + |11\rangle}{\sqrt{2}}$$

This procedure creates the Bell state $|\Phi^+\rangle$ regardless of measurement outcome, demonstrating how measurement plus conditional corrections can generate entanglement from separable states.

3.4 Decoherence in multi-qubit systems

When dealing with multiple qubits, decoherence becomes even more challenging than in single-qubit systems. Not only can each qubit experience individual decoherence, but the interactions between qubits can create new pathways for errors. The main types of multi-qubit decoherence are:

1. **Independent decoherence**: Each qubit experiences its own local noise
2. **Correlated decoherence**: Environmental effects that simultaneously affect multiple qubits
3. **Cross-talk**: Unwanted interactions between qubits that should be isolated

For example, consider a two-qubit state that starts in a Bell state:

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$

The density matrix for this pure state is:

$$|\Phi^+\rangle\langle\Phi^+| = \frac{1}{2}\begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix}$$

Under independent dephasing, each qubit loses phase coherence separately:

$$\rho(t) = \frac{1}{2}\begin{pmatrix} 1 & 0 & 0 & e^{-2\gamma t} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ e^{-2\gamma t} & 0 & 0 & 1 \end{pmatrix}$$

which would represent exponential decay in entanglement over time; this could occur even with pure local noise. This highlights a crucial feature of quantum systems: even when noise acts independently on each qubit (i.e., local noise), it can destroy global quantum properties like entanglement. In other words, we don't need correlated noise to degrade entanglement - local noise channels are sufficient to compromise the quantum advantages that entanglement provides.

Example: Impact of different noise types

Consider a Bell state under different noise channels:

1. Amplitude damping on first qubit only:

$$\rho(t) = \begin{pmatrix} 1 - \frac{e^{-\gamma t}}{2} & 0 & 0 & \frac{e^{-\gamma t/2}}{\sqrt{2}} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \frac{e^{-\gamma t/2}}{\sqrt{2}} & 0 & 0 & \frac{e^{-\gamma t}}{2} \end{pmatrix}$$

2. Dephasing on both qubits:

$$\rho(t) = \frac{1}{2} \begin{pmatrix} 1 & 0 & 0 & e^{-2\gamma t} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ e^{-2\gamma t} & 0 & 0 & 1 \end{pmatrix}$$

This shows how different noise channels affect the quantum correlations in distinct ways.

3.4.1 Gate Errors

In addition to decoherence during idle times, errors can occur during gate operations. These gate errors come in several forms:

1. **Systematic errors**: Consistent over/under-rotation of gates
2. **Random errors**: Fluctuations in gate parameters
3. **Cross-talk errors**: Gates affecting neighboring qubits
4. **Leakage errors**: System leaving the computational basis

For two-qubit gates like CNOT, errors are typically higher than single-qubit gates because:

- They require stronger interactions between qubits
- Take longer to implement
- Are more sensitive to timing and control errors (e.g.,some qubits have to wait while single-qubit operations “catch up”).

Current State of Gate Fidelities

As of 2021, record error rates are:

- Single-qubit gates: 0.03%
- Two-qubit gates: 0.5%
- Measurement: 0.2%

These numbers are representative of state-of-the-art superconducting qubit platforms, as reported in [28].

References

[1] F. Arute et al., Quantum supremacy using a programmable superconducting processor, Nature **574**, 505 (2019).

[2] Google Quantum AI and Collaborators et al., Quantum error correction below the surface code threshold, Nature (2024).

[3] E. Pednault, J. A. Gunnels, G. Nannicini, L. Horesh, and R. Wisnieff, Leveraging Secondary Storage to Simulate Deep 54-qubit Sycamore Circuits, arXiv:1910.09534 (2019).

[4] F. Pan, K. Chen, and P. Zhang, Solving the Sampling Problem of the Sycamore Quantum Circuits, Physical Review Letters **129**, 090502 (2022).

[5] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition*, Anniversary edition (Cambridge University Press, Cambridge ; New York, 2011).

[6] A. M. Turing, On Computable Numbers, with an Application to the Entscheidungsproblem, Proceedings of the London Mathematical Society **s2-42**, 230 (1937).

[7] R. Horodecki, M. Horodecki, and K. Horodecki, Quantum entanglement, Reviews of Modern Physics **81**, 865 (2009).

[8] L. Hoddeson, The Discovery of the Point-Contact Transistor, Historical Studies in the Physical Sciences **12**, 41 (1981).

[9] R. P. Feynman, Simulating physics with computers, International Journal of Theoretical Physics **21**, 467 (1982).

[10] D. Deutsch, Quantum theory, the Church–Turing principle and the universal quantum computer, Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences **400**, 97 (1985).

[11] D. Deutsch and R. Jozsa, Rapid solution of problems by quantum computation, Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences **439**, 553 (1992).

[12] E. Bernstein and U. Vazirani, Quantum Complexity Theory, SIAM Journal on Computing **26**, 1411 (1997).

[13] D. R. Simon, On the Power of Quantum Computation, SIAM Journal on Computing **26**, 1474 (1997).

[14] P. W. Shor, *Algorithms for Quantum Computation: Discrete Logarithms and Factoring*, in *Proceedings 35th Annual Symposium on Foundations of Computer Science* (IEEE Comput. Soc. Press, Santa Fe, NM, USA, 1994), pp. 124–134.

[15] L. K. Grover, *A Fast Quantum Mechanical Algorithm for Database Search*, in *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing - STOC '96* (ACM Press, Philadelphia, Pennsylvania, United States, 1996), pp. 212–219.

[16] W. K. Wootters and W. H. Zurek, A single quantum cannot be cloned, Nature **299**, 802 (1982).

[17] S. Bravyi and A. Kitaev, Universal quantum computation with ideal Clifford gates and noisy ancillas, Physical Review A **71**, 022316 (2005).

[18] M. M. Wilde, *Quantum Information Theory*, 2nd ed (Cambridge university press, Cambridge, 2017).

[19] E. Schrödinger, Discussion of Probability Relations between Separated Systems, Mathematical Proceedings of the Cambridge Philosophical Society **31**, 555 (1935).

[20] A. Einstein, B. Podolsky, and N. Rosen, Can Quantum-Mechanical Description of Physical Reality Be Considered Complete?, Physical Review **47**, 777 (1935).

[21] J. S. Bell, On the Einstein Podolsky Rosen paradox, Physics Physique Fizika **1**, 195 (1964).

[22] S. J. Freedman and J. F. Clauser, Experimental Test of Local Hidden-Variable Theories, Physical Review Letters **28**, 938 (1972).

[23] A. Aspect, J. Dalibard, and G. Roger, Experimental Test of Bell's Inequalities Using Time-Varying Analyzers, Physical Review Letters **49**, 1804 (1982).

[24] M. F. Pusey, J. Barrett, and T. Rudolph, On the reality of the quantum state, Nature Physics **8**, 475 (2012).

[25] N. D. Mermin, Bringing home the atomic world: Quantum mysteries for anybody, American Journal of Physics **49**, 940 (1981).

[26] W. Wootters, Entanglement of formation and concurrence, Quantum Information and Computation **1**, 27 (2001).

[27] D. M. Greenberger, M. A. Horne, A. Shimony, and A. Zeilinger, Bell's theorem without inequalities, American Journal of Physics **58**, 1131 (1990).

[28] N. P. De Leon, K. M. Itoh, D. Kim, K. K. Mehta, T. E. Northup, H. Paik, B. S. Palmer, N. Samarth, S. Sangtawesin, and D. W. Steuerman, Materials challenges and opportunities for quantum computing hardware, Science **372**, eabb2823 (2021).