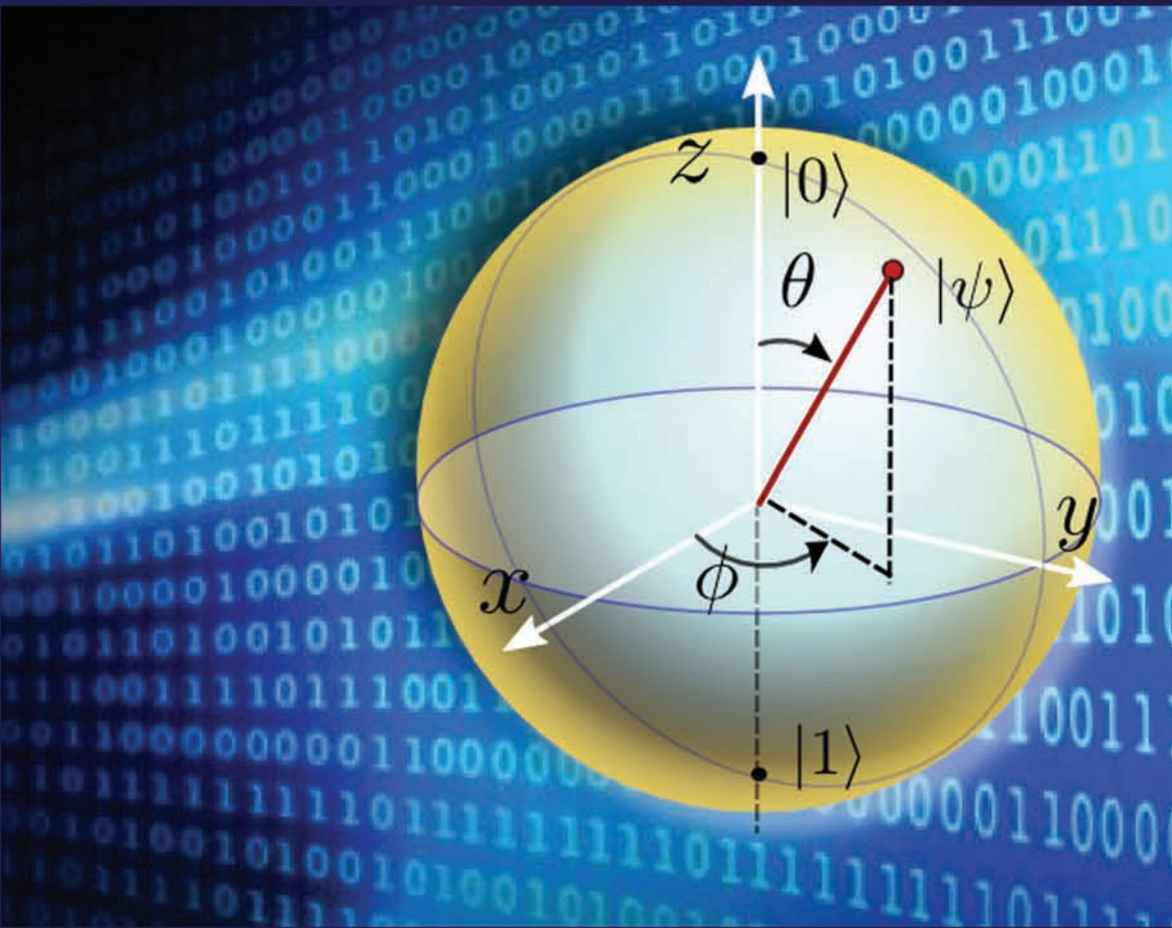
INTRODUCTION TO QUANTUM PHYSICS AND INFORMATION PROCESSING



Radhika Vathsan

CRC Press
Taylor & Francis Group

INTRODUCTION TO QUANTUM PHYSICS AND INFORMATION PROCESSING

INTRODUCTION TO QUANTUM PHYSICS AND INFORMATION PROCESSING

Radhika Vathsan
BITS Pilani K.K. Birla Goa Campus
India

CRC Press
Taylor & Francis Group
Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business
A CHAPMAN & HALL BOOK

CRC Press
Taylor & Francis Group
6000 Broken Sound Parkway NW, Suite 300
Boca Raton, FL 33487-2742

Version Date: 20150714

International Standard Book Number-13: 978-1-4822-3812-9 (eBook - PDF)

Visit the Taylor & Francis Web site at
http://www.taylorandfrancis.com

and the CRC Press Web site at
http://www.crcpress.com

Contents

Preface

This book is aimed at undergraduates or beginning graduates in physics, mathematics or engineering, as an introduction to the developing field of quantum information. The book has grown out of lecture notes for a one-semester course offered to undergraduate students of engineering and science at the BITS Pilani Goa Campus. The level is consequently basic, and is intended to train a student with no background in quantum physics to be able to follow more advanced books on the subject, such as the classic by Neilsen and Chuang, and also research material in this rapidly growing field. We start from the basics of quantum mechanics required to understand two-level systems to be used as qubits, and then go on to show how quantum properties are exploited in devising algorithms for problems that are more efficient than the classical counterpart. We then go on to more sophisticated notions that form the backbone of quantum information theory.

The idea behind a book at this level is that a student doesn't need to go through a masters course in physics to get started on quantum information, an interdisciplinary subject that is currently in an exciting stage of development. The treatment of the subject matter is elementary but with a bias toward ideas of foundational importance in quantum information science. It consequently goes deeper into issues of understanding quantum theory without raising the technical level too much. Fully aware of being perhaps too wordy, some of the introductory material is described at length to impress upon the student the true meaning of the quantum mechanical description of nature.

The book is presented in four parts. The first preliminary part sets the stage by introducing the methods and notation of quantum mechanics of finite state systems. We begin with a thorough but brief description of a typical two-state system: electron spin, using the Stern–Gerlach system as an illustrative medium. The second part sets the theoretical framework in place, starting with the rules of quantum mechanics in the language of linear algebra. With a view to completing the background required to understand current research papers, we have also included a slightly advanced chapter on the density matrix approach to the characterization of mixed states and open systems. We also include a brief on concepts in computer science such as the circuit model for computation, computational complexity, and reversible computation, after the manner of Neilsen and Chuang.

The third part deals with quantum computation, starting with universal quantum gates and circuits. We treat the basic quantum algorithms such as the Deutsch–Josza, Grover and Fourier transform-based algorithms, which are discussed in some detail. The fourth part on quantum information addresses the notion of information content in qubits, cryptographic applications of quantum information processing, and quantum error correction. We have also included a chapter on slightly advanced material dealing with the characterization of quantum information, to bridge the gap in the material undergraduate students are normally exposed to, and current research literature in quantum information theory.

Richly illustrated by examples and supplemented by exercises and problems, the book is intended to take the beginner seamlessly to the state of current research in the area, so that the advanced literature in this fast-developing field can be easily followed. The reader is led through example and detailed discussions to understanding some of the deeper concepts of quantum theory that can be put to use in this area of the subject. Each chapter is accompanied by pointers to references that take the reader beyond what is presented in the book.

Acknowledgments

A work of this nature is simply not possible without the inspiration, guidance, help, and cooperation of numerous people. I would first like to dedicate this book to the memory of Prof. Suresh Ramaswamy whose encouragement was the inspiration to start on this project. Thanks are due to my faculty colleagues who provided guidance, to the many enthusiastic students who took my course on quantum information and computation at the BITS Pilani Goa Campus, which formed the base on which this book was built, and to the kind cooperation of the Raman Research Institute, Bangalore, for extending its hospitality during a sabbatical period in 2013 during which part of this work was completed.

I would like to individually thank my students Aarthi Sundaram, Anupam Mitra, Arunav Sanyal, Bharath Srivathsan, Mayank Jog, Milind Shyani, Nisheeth Bandaru, Pratik Mallya, Shubhanshu Tiwari, Tarik Gupta, and many others for vital class discussions that made me eagerly look forward to the pleasure of their class, and for helping produce solutions to problems. I'd like to thank my colleagues Andal Narayanan, Anu Venugopalan, Tabish Qureshi, Joseph Samuel, Supurna Sinha, A. R. Usha Devi, Sumati Surya and Urbasi Sinha, for stimulating discussions. Thanks are due to Dhavala Suri, a graduate student in our department, for proofreading the material and offering to help with the clarity and effectiveness of the delivery.

I am greatly indebted to the creators and improvers of the fantastic typesetting environment of LaTeX, which was one of the inducements for writing this book! Referring to Leslie Lamport's *LaTeX* and Mittelbach and Goossen's

The LaTeXCompanion made designing the book layout great fun. The quantum circuit diagrams in this book owe their neatness and ease of typesetting to the wonderful LaTeXpackage QCircuit designed by Steve Flammia and Bryan Eastin. I owe a lot to the Inkscape vector graphics package for making the illustrations a pleasure to design and draw.

I am indebted to the CRC team starting with my editor, Aastha Sharma, for encouraging me to bring these notes out in book form, to the anonymous referees for insightful comments, and to Karen Simon for the fine editing.

Finally, I owe much to my mother, who inculcated in me a sense of perfectionism, aesthetic sensibility, and a never-say-die attitude (and provided a lot of training in copy editing!), my father who introduced me to Feynman's legendary lecture notes at a young age and spurred my dream of being a theoretical physicist, my sister whose sense of humor and support carried me through many dark moments, and of course to Kshipra who put up with my many moods and late timings while amma's book was being put together.

List of Figures

List of Tables

List of Boxes

List of Symbols and Notations

z^*	Complex Conjugate, $i \to -i$ of a complex number z
$\|\psi\rangle$	A general unknown quantum state vector
$\|0\rangle$	Single qubit basis state, positive eigenstate of σ_z
$\|1\rangle$	Single qubit basis state, negative eigenstate of σ_z
$\|x\rangle$	A generic computational basis state
$\hat{A}$	An operator, also represented as a matrix
$\hat{A}^{\mathrm{T}}$	Transpose of the matrix A, obtained by interchanging the rows and columns
$A^\dagger$	Adjoint, or complex conjugate transpose $(A^*)^{\mathrm{T}}$ of a matrix A
$[A, B]$	Commutator $AB - BA$ of the given matrices
$\{A, B\}$	Anti-commutator, $AB + BA$ of the given matrices
$\mathrm{Tr}(X)$	Trace of matrix X
$\mathrm{Tr}_A(X)$	Partial trace over system A
$\mathcal{P}(r)$	Probability of occurrence of value r
$\hat{\mathbb{P}}_i$	Projection operator along the direction $\|i\rangle$
$\sigma_x, \hat{X}$	Pauli X-matrix
$\sigma_y, \hat{Y}$	Pauli Y matrix
$\sigma_z, \hat{Z}$	Pauli Z matrix
$\oplus$	XOR or addition mod 2
$\mathbb{1}$	Identity matrix or operator
$\vec{\boldsymbol{\sigma}}$	Collective of the three Pauli matrices
$\vec{\boldsymbol{a}} \cdot \vec{\boldsymbol{\sigma}}$	The linear combination $a_1\sigma_x + a_2\sigma_y + a_3\sigma_z$ for any three numbers $\{a_1, a_2, a_3\}$

Commonly used notation

Computational basis states:
$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

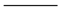
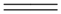
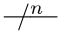
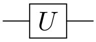
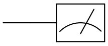
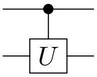
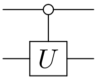
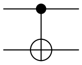
Bell basis states:
$$|\beta_{00}\rangle = \tfrac{1}{\sqrt{2}} (|00\rangle + |11\rangle)$$
$$|\beta_{01}\rangle = \tfrac{1}{\sqrt{2}} (|00\rangle - |11\rangle)$$
$$|\beta_{10}\rangle = \tfrac{1}{\sqrt{2}} (|01\rangle + |10\rangle)$$
$$|\beta_{11}\rangle = \tfrac{1}{\sqrt{2}} (|01\rangle - |10\rangle)$$

Pauli matrices/gates: $Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}.$

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

Phase gate: $\phi = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\phi} \end{bmatrix}.$

Quantum circuit notation

A quantum wire	
A classical wire	
An n-qubit register	
A quantum gate U	
Measurement	
A 1-controlled-U gate	
A 0-controlled-U gate	
The CNOT gate	

About the Author

Dr. Radhika Vathsan earned a PhD in mathematical physics and is currently an associate professor in the Department of Physics of the Goa campus of Birla Institute of Technology & Science, Pilani. After completing an honours degree in physics from St Stephen's College, Delhi University for her bachelor's degree, she obtained a first class in her masters in physics from the Indian Institute of Technology, Madras in 1992. Her PhD was from the University of Madras, after studying at the Institute of Mathematical Sciences, Chennai. She spent two years at the Harish Chandra Research Institute at Allahabad before joining the faculty of physics of BITS Pilani at their Pilani Campus in 2002.

While her interests range from formal mathematical techniques of quantization to quantization of many-body systems, she has now branched out to foundations of quantum theory and quantum information. A popular teacher and inspiring lecturer, she is also a Veena player with an abiding interest in Indian classical music.

Part I

Preliminaries

Chapter 1

Introduction

Information as accessed by the human mind, and the myriad ways of processing it, is what has set humankind apart in the animal world. The ability to use the physical systems around us to encode and then to process information has been evolving in leaps and bounds ever since the dawn of our race! The earliest form of computation was probably in the form of account-keeping by counting pebbles or on the fingers. This developed into the abacus, writing symbols, the computing machine, and now, a laptop or a supercomputer: everywhere we represent information by means of a *physical* system, and perform manipulations on that system to process the information in many desired ways, including communication. The stress here is on the realization that the basis of information is a physical system. The more advanced that system is and the set of rules it functions on, the more capable our means of information processing and communication. When the underlying physical system used for encoding and processing information is a *quantum* system, we have quantum information processing.

While technological advances have made it possible to reach astounding speeds and processing power, the basic paradigm of current day information processing is binary logic with currents or voltages in the semiconductor circuitry at the heart of the modern computer processors. However, it is important to realize that the behavior of these high and low states of the circuit is based on laws of classical physics. We know now that at the most fundamental level, physical systems obey the laws of quantum mechanics. These laws are fundamentally different in many ways from classical laws of physics. Therefore, the basic paradigm of information processing is different when we come to quantum information processing. Not only are the algorithms and the processing mechanism different, but there are distinct advantages of the quantum over the classical.

According to recent data, the fastest current day supercomputer is capable of performing at a speed of hundreds of Gflop/s.[1] The quantum paradigm affords a speedup to many algorithms that are very slow to perform even on this computer! Much of modern-day information security, for instance secure online cash transactions, is based on classical cryptography. This has been proved to be vulnerable if a quantum computer is used to crack the code!

I've been trying to motivate the need to study quantum information. But

[1] Gigaflop per second, $Giga = 10^{12}$, *FLOP = floating point operation.*

the inquisitive scientific mind will no doubt want to grasp the basics of the physical laws that make complex information processing possible: the laws of quantum mechanics. Professor Richard Feynman of Caltech was supposed to have famously said that *no one understands quantum mechanics!* How then are we basing modern technology on it? With this book I'd like to show you that despite its "weirdness", by which I mean its distance from our common sense understanding which appears wired to classical physics, the laws of quantum mechanics can be apprehended by an undergraduate student, to be used as a set of rules by which the game is played. The more philosophically inclined will be drawn to ponder meaning and interpretation of these rules. And this latter exercise is also rewarding, in bringing out fascinating new facets of quantum theory, to be exploited in our ever-expanding game of information processing.

Several considerations make the transition to the quantum inevitable while exploring efficient information processing. One is from the perspective of hardware engineering, where miniaturization and the need to pack more structure in less space must eventually lead to the limit set by the structure of matter: the atomic or even electronic level. At this level, classical laws of physics are no longer valid and we have to consider the essentially quantum nature of the physical system used to store and manipulate information.

However, from the angle of the basic physical laws of quantum mechanics, more complex ways of processing information should be possible. The manner in which a quantum system evolves, transforms information and conveys it in an experiment via a measurement, is fundamentally different from classical information. This was realized first by Feynman [33] in the 1980s when he pointed out that a quantum process cannot be efficiently simulated on a classical computer. He showed, however, that such a system may be efficiently simulated on a *quantum* computer.

In the process of studying how this is possible, we are led into a deeper probing of the foundations of quantum physics. In implementing a quantum computer, physicists need to access and control *individual* quantum states, prepare them, manipulate them and finally, measure them. The question also arises of how to deal with practical systems that are not ideal and isolated from their environments, but are subject to noise or errors due to inadvertent environmental effects. These considerations lead us into an experimental regime of testing our ideas of quantum reality, and into discovering new quantum phenomena.

The third perspective is from the theory of computation. The foundations of modern computer science may be said to have been laid by the work of Alan Turing in the 1930s [69], on abstract models of computing embodied in what is now known as the *Turing Machine.* The *Universal Turing Machine* (UTM) is an idealization of a model of computation that can execute any computable algorithm, in short, any task that can be run on a modern programmable computer. Notions of computability of problems and efficiency of algorithms were developed. In rough terms, an algorithm is said to be efficient if it takes *polynomial time* for execution. This means that the time required to run it grows as a polynomial in the number n of input bits, i.e., at most as a power of n. A computationally *hard* problem is one for which the best algorithm is exponential in the number of input bits, i.e., grows as a^n where a is some constant.

As Feynman pointed out, the evolution of a quantum system is one prime example of a problem that cannot be simulated efficiently on a Turing machine, while a quantum computer could quite naturally do so! This was a challenge to the Church–Turing thesis that any computation can be efficiently simulated on a Turing Machine. This thesis had been modified to include probabilistic machines (based on fuzzy logic) but now has been extended to a quantum version.

Problems in computational complexity have now been extended to include the quantum Turing machine and the possibilities are exciting. While the basic notion of computability of a given problem does not change when quantum machines are included, problems that were hard classically may become easy. Quantum computation may also resolve other questions in the area of computational complexity.

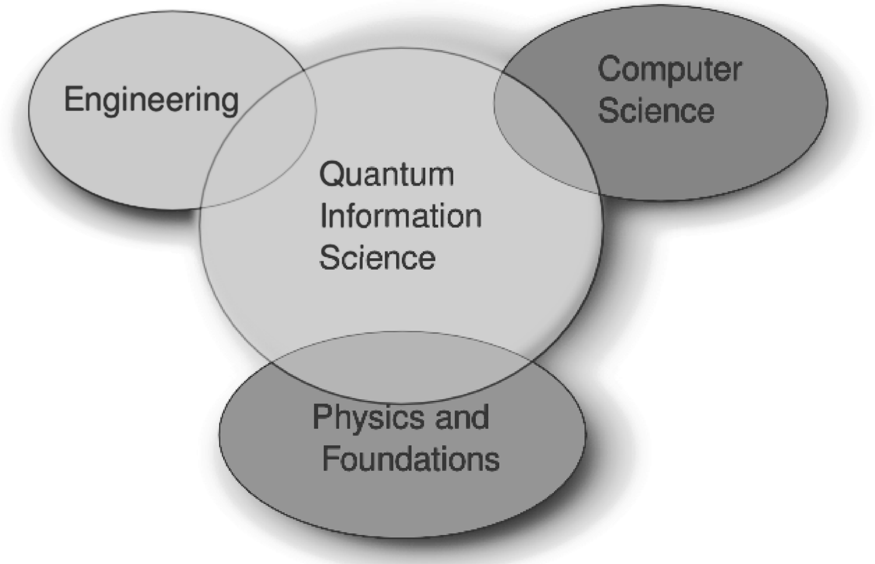


FIGURE 1.1: Three approaches to the quantum.

These three approaches (Figure (1.1) have historically motivated the study of this field, which at present however, is rapidly blooming in several different directions unforeseen in the last century.

1.1 Bits and Qubits

So what is the fundamental difference between classical and quantum computing?

Computation and information processing as we know it today is built upon Boolean logic and algebra. Boolean algebra is binary, requiring two logical units called bits. You can think of them as the possible answers to a decision question: yes or no. The idea is that almost any problem can be reformulated as a series of decision questions, and therefore can be encoded in bits. A *bit*, or binary digit, is a physical system that can take on two logical states represented by 0 and 1. In a typical digital computer these states are the low and high voltage states in the microcircuitry.

To extend the capabilities of the computational system, probabilistic algorithms are based on the notion of fuzzy bits, that can take the value 0 with a probability p or 1 with probability $1 - p$. This is the basis of probabilistic computation, or so-called fuzzy logic.

When the logic is extended to binary states of a quantum system, we arrive at the *qubit*, a quantum bit. A qubit can take values 0 or 1 but with probabilities given by the mod-squared of complex numbers! A physical qubit is a quantum system that will represent our Boolean units. The system can therefore take on two quantum states that we will now represent in the notation $|0\rangle$ and $|1\rangle$, to distinguish them from the states 0 and 1 of the classical bit. This angular bracket notation is due to physicist Paul Dirac [29]. This notation is very versatile and by itself a useful calculational tool.

A qubit is generically represented as a linear *superposition* of the basis states:

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

The coefficients α and β are called *probability amplitudes*, and satisfy such that $|\alpha|^2 + |\beta|^2 = 1$.

The way to understand this statement is that upon measurement, the generic qubit takes on one of the definite states $|0\rangle$ or $|1\rangle$ with a probability $|\alpha|^2$ or $|\beta|^2$. In this sense, a qubit is similar to a classical bit in that measurement only gives one of two values. It is sometimes useful to think of these values, or the basis states of the qubit, as classical bits.

Though the qubit is probabilistic, it differs from the fuzzy bit because of the possibility of interference. This is characteristic that is captured by the complex amplitudes. A complex number has a magnitude and well as a phase. While composing two or more such numbers, the phases could result in reinforcement or reduction in the strength of the resultant. The physical implications of this is familiar to us through the phenomenon of interference in optics. When two beams of light, described by electric fields having definite phase relationships to each other, are combined, then there are regions

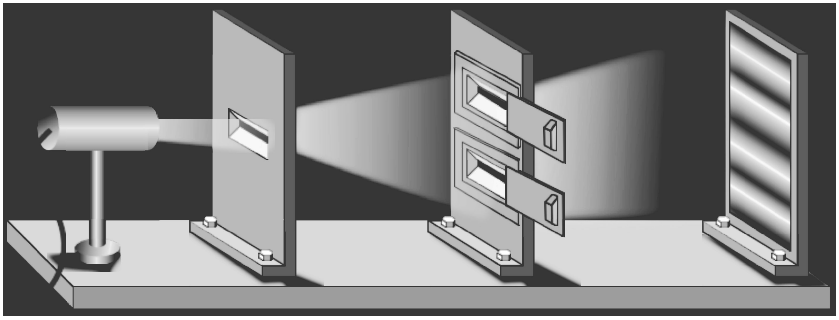


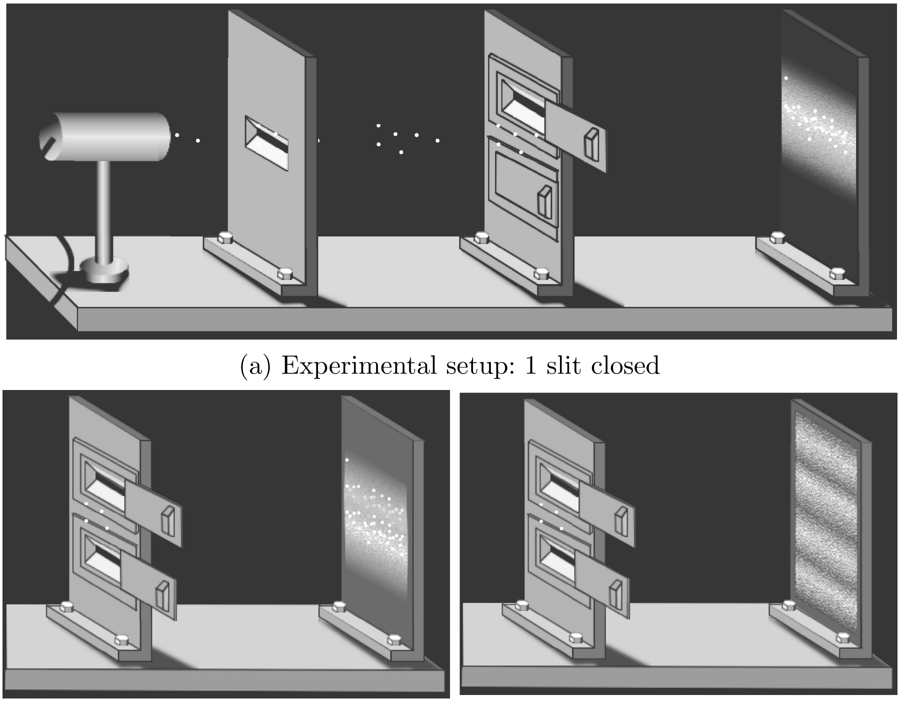
FIGURE 1.2: Two-slit interference with light.

where the light cancels out (destructive interference) and regions where it gets reinforced (constructive interference). The paradigmatic example is the dark and bright fringes formed when two slits are illuminated by monochromatic (coherent) light (Figure 1.2)[2].

The same phenomenon is observed when two slits are "illuminated" by a stream of quantum particles (Figure 1.3). The particles hit the screen at different positions, and intensity of the pattern on the screen is interpreted as the probability of a particle striking the screen at that position. Classically, we would have expected peaks of intensity directly behind the two slits. The only way of explaining the alternating fringes of maximum and minimum intensity is to consider the states of the particles exiting the two slits as described by complex amplitudes. The interference between these amplitudes perfectly predict the fringe pattern. The general superposition state (Equation 1.1) is therefore aptly called a *coherent superposition.*

One can visualize the computational space afforded by a single classical bit as two discrete points, that is a zero-dimensional space. For a fuzzy bit, it is all points on a line segment $[0, 1]$, one dimensional. For a quantum bit, which is like a unit vector that can point in any direction, the representation is any point on a unit sphere, and the space is two-dimensional. We will see how this is so in the coming chapters. The little caricature of Figure 1.4 captures the essential difference in the computational capability of these bits.

[2] The apparatus in these figures has been drawn in a semi-realistic fashion after Bohr [14], in his famous arguments with Einstein about the complementarity between wave-like and particle-like behavior of quantum objects



(a) Experimental setup: 1 slit closed

(b) Classically expected result

(c) Actual result: interference

FIGURE 1.3: Two-slit experiment with quantum particles.

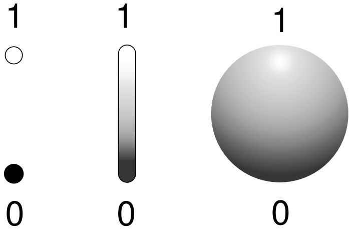


FIGURE 1.4: Visualizing a classical bit, a fuzzy bit and a quantum bit.

1.2 Properties of Qubits

Quantum systems have certain properties that are counter-intuitive and completely outside our range of experience in the classical world. These "weird" properties are best understood as inevitable consequences of axioms on which quantum laws are based. These axioms have been arrived at after considerable effort and study of experimental phenomena, and are now accepted among physicists as the complete theory which describes the real world at the fundamental level.

The basic mathematical properties of a qubit can be analyzed and studied independent of the physical system that realizes it. By treating the qubit as an abstract mathematical entity, we can develop a general theory of quantum information processing. Some of the strange new properties that become relevant are now discussed.

Superposition and quantum parallelism: The main implication of states like that of Equation 1.1 is that a single state contains the potential for the system to be in either basis state. In some sense the system, say an electron characterized by its spin value, simultaneously exists in both states until measured. Physically this does not seem to make sense to our classical minds unless we say that the electron has not decided which of the two possible states it should be in, until forced into one of them by a measurement.

This feature is exploited in quantum computation to implement what is called *quantum parallelism*: an operation that acts on a bit can now simultaneously act on both possible values of the bit if the input is a qubit in a quantum superposition.

Size of computational space: If we want to do an n-bit computation, Classically the "space" available for computation is of size n. In terms of a quantum system of n qubits, the number of possible basis states is 2^n, and this is the size of the space available for computation. The size of the space of states available for computation grows exponentially with the number of bits (Figure 1.5). This is the power we wish to exploit in quantum computation.

Entanglement and quantum correlations: Multiple qubit systems can exist in superposition states that are known as *entangled* states. These states possess intrinsic correlations between the component systems that are different from classical correlations. These correlations can survive even if the component systems are taken physically far apart from each other. For example, 2-qubit states are in general linear superpositions of $|00\rangle, |01\rangle, |10\rangle$, and $|11\rangle$. Look at the state $\frac{1}{2}(|00\rangle + |11\rangle)$. In such a state, the first and second systems are correlated quantum mechanically: the value of the second qubit is always equal to that of the first qubit, irrespective of what measurement we make on which bit and when. Such a state is called "entangled" because of this correlation.

Quantum correlations can be exploited to generate new methods of pro-

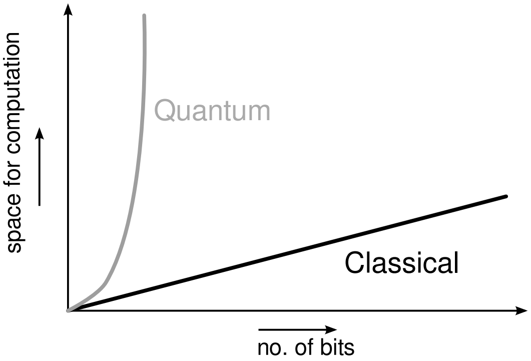


FIGURE 1.5: Computational power: quantum vs classical.

cessing, increasing the efficiency by allowing controlled operations to be performed. These correlations are an invaluable resource in quantum information theory and we will see their basic applications in quantum state teleportation and secure information transfer over a distance.

Measurement and state collapse: Though a qubit could exist in a superposition of basis states, a measurement of the qubit would give one of the two basis states alone. Measurement of a quantum system causes it to collapse into one of the basis states, which destroys the superposition, including any information that may be encoded in the probability amplitudes. Some authors express this property as a qubit existing in a superposition not having a definite state. Measurement results can be predicted with 100% certainty in "definite" states, and the system exists in a basis state. When a system is not in a definite state, measurement disturbs the system and one can never know the original state exactly. It is a quantitative and in-depth study of quantum measurements that has uncovered new laws of quantum information.

Unitary evolution and reversibility: Quantum dynamical laws governing the evolution of an isolated quantum system are what are known as *unitary* evolutions. Thus the functioning of a quantum computer is necessarily via unitary transformations of the initial quantum state. Unitary operations are fully reversible and, from a large body of study on the energetics of computation, are said to lead to greater energy efficiency.

No cloning: This is another peculiar property of generic quantum states: quantum states that are not basis states cannot be perfectly cloned or copied. The fact that classical states can be copied and kept aside for further processing is often taken for granted. When implementing a function in a classical circuit, we often send copies of a particular input to different parts of the circuit. Such an operation is no longer possible in quantum computing. This changes the way we look at a quantum computation. And on the upside, this also makes it possible to exchange information in a secure manner since tapping a quantum line disturbs the system irrevocably!

These properties lead us naturally to a model of computation often called the “circuit model,” based on classical logic-gate circuits, of quantum computation, which is what we will primarily study in this book. However, several other models of quantum information processing have also evolved, such as measurement-based computation, continuous-variable computation and adiabatic evolution. The interested reader may refer to the literature for these.

1.3 Practical Considerations

Theoretically, the examination of the paradigm of quantum computation has been very promising and exciting. However these considerations need to be grounded in reality. Pure quantum systems are found at the microscopic scale and are difficult to access except by special technological means. To initialize any information process, we must have the means to assign any desired state to the qubit. Manipulation of the states of an individual qubit requires a high level of technological ingenuity. We need not just one qubit but large qubit registers. These may be built out of a collection of non-interacting qubits but whether such a register can be built for the system at hand brings in questions of scalability.

In implementing a quantum gate, we would be required to assemble some means to applying forces on the system in a precise and accurate manner. These operations would have to be impervious to error. The major problem in practice with quantum superposition states is that they are extremely fragile. The slightest interactions would cause a disturbance by which the coherence is lost and the prepared system ends up in one of the basis states! This phenomenon, known in literature as *decoherence*, is also crucial in understanding how the classical world emerges from the quantum substrate. However, the discovery of quantum error correction and the subsequent construction of fault-tolerant computing has infused confidence in the success of the paradigm despite this issue.

The final big challenge is in interpreting the results of a measurement on the system. The whole computational process must be set up such that the end result is one of the basis states so that measurements give definite and not probabilistic outcomes.

It may indeed be justifiable to ask if quantum computation is just in theory, a matter of fanciful speculation, or possible in concrete implementation. While there are technical challenges in the building of a feasible quantum computer, the actual implementation is not only possible but also a reality. Various ingenious techniques in quantum physics have been implemented, and newer ones are being rapidly developed.

In developing a viable physical implementation, a bunch of criteria, first to be underlined by DiVincenzo [30], are to be satisfied:

1. A robust, error-tolerant system for qubits
2. A method of initializing (preparing initial states)
3. Scalability: quantum systems that must be replicated to larger numbers to make bigger registers
4. Ability to manipulate individual quantum states: this is the most challenging engineering task that is required to make the computer work
5. Readout of output: the end result of the computation must be readable, that is, measurement with unambiguous results.

Several systems have been analyzed with these criteria in mind. In a given system too, there could be different possible realizations of a qubit. In Table 1.1, we list a few such systems to give you an idea of the variety in the physics that is involved.

TABLE 1.1: Summary of common physical implementations of quantum computing systems.

System	**Information carrier**	**Method of control**
Quantum Optics	photon polarization presence of a single photon in one of two modes	polarizers, half wave plates, quarter wave plates beamsplitters, mirrors, and non-linear optical media
Cavity QED	two-level atom interacting with a single photon	phase-shifters, beam splitters, and other linear optical elements
Trapped Ions	hyperfine energy levels and the vibrational modes of the atom	pulsed laser light to manipulate the atomic state
Nuclear Magnetic Resonance (NMR)	nuclear spin states	pulsed RF fields in the presence of a strong external magnetic field
Superconducting Circuits	Cooper-pair box flux-coupled SQUID current-biased junction	electrostatic gates and Josephson junctions magnetic fields, spin interactions pulsed microwave fields
Quantum Dots	electron spin charge state	magnetic fields and voltage pulses electrostatic gates and waveguides

It is into this amazingly rich and novel quantum world that we are going plunge now, starting at the surface, to get a broad overview and an understanding of the fundamentals.

1.4 References for Further Reading

Much of the material in this book is based on some of the classics on the subject: Nielsen and Chuang [50], the encyclopedic tome which was the first textbook on quantum computation when the field was mainly one for advanced research; Mermin [48] which is an excellent introduction to quantum computing for the non-specialist; John Preskill's online lecture notes from his Caltech classes [57]; and for more subtle ideas in quantum physics, Peres [54] and other references that are cited at the relevant context. Other wonderful books introducing this subject, at a slightly more advanced level, are by Yanofsky and Mannucci [77]; Rieffel and Polak [58] and Stolze and Suter [67]. One of the first books written for students on this subject, by Williams and Clearwater [74] has various new insights and computer simulation exercises that will be useful to a newcomer. Further references to original work and review articles will be given in the specific chapters at the relevant places.

Chapter 2

A Simple Quantum System

To obtain a basic understanding of how and why quantum computing works, one needs to understand quantum theory. In this book we treat quantum mechanics as an axiomatic theory, with some interpretations where possible. The implications of quantum mechanics are often counter intuitive, or rather, beyond intuition, since we have no direct experience of the quantum world as we do the classical. We assume no prior knowledge of quantum mechanics, and will develop all the basic tools and ideas necessary for a sound grasp of the fundamentals of the theory, especially what is relevant for two-state systems that we'll use as qubits. For a more detailed picture, the reader is referred to standard texts on quantum mechanics, such as the books by J. J. Sakurai [60], R. Shankar [62], C. Cohen-Tannoudji [20], E. Merzbacher [49] etc. or the Feynman Lectures on Physics Vol III [34].

The language of the science of describing the physical world is mathematics: universal, unambiguous, and precise. Without this language it is not possible to express correctly the properties of the physical world, nor is it possible to predict new physical phenomena that can then be examined by experiments. Thus it is imperative that we learn to speak and write this language. The mathematical language of quantum mechanics is that of vector spaces and linear algebra. States of an isolated quantum system are accurately designated as vectors in a complex vector space known as *Hilbert Space.* The evolution of the system with time, or under the influence of forces is expressed as linear operators acting on these vectors.

Our aim is to understand how a quantum system can be used to carry information. We wish to identify a system that can exist in a discrete number of distinct states, in fact two states, to form a qubit. We will try to put the mathematics in context by first examining the structure of a simple 2-state quantum system via experiments that bring out the basic nature of qubits, and the need for this new language to communicate their properties.

2.1 The Stern–Gerlach Experiment

Way back in 1922, when physicists were still studying the new and astonishing properties of the basic constituents of matter, an experiment designed to measure the magnetic moment of atoms gave unexpected results. This was the classic Stern–Gerlach experiment [36], designed to measure the magnetic moments of atoms. The results brought out a new quantum property of an electron, called intrinsic spin, which could take on quantized values, i.e., one of two values only.

We can get a feel for the physics by looking at the classical definition of the magnetic moment. The revolution of electrons around the nucleus of an atom is like a circulating current, and a circulating current is a *magnetic dipole.* The dipole moment $\vec{\boldsymbol{\mu}}$ equals the current times the area of the current loop, with direction given by the axis about which the current circulates. When a magnetic dipole is subjected to a non-uniform magnetic field $\overrightarrow{\boldsymbol{B}}(\vec{\boldsymbol{r}})$, it feels a force along the direction of *change* of the field: $\overrightarrow{\boldsymbol{F}} = \overrightarrow{\boldsymbol{\nabla}}(\vec{\boldsymbol{\mu}} \cdot \overrightarrow{\boldsymbol{B}})$, and will be deflected. Measuring the deflection in a known magnetic field, the value of the magnetic moment can be calculated.

A schematic setup of the Stern–Gerlach type is shown in Figure 2.1.

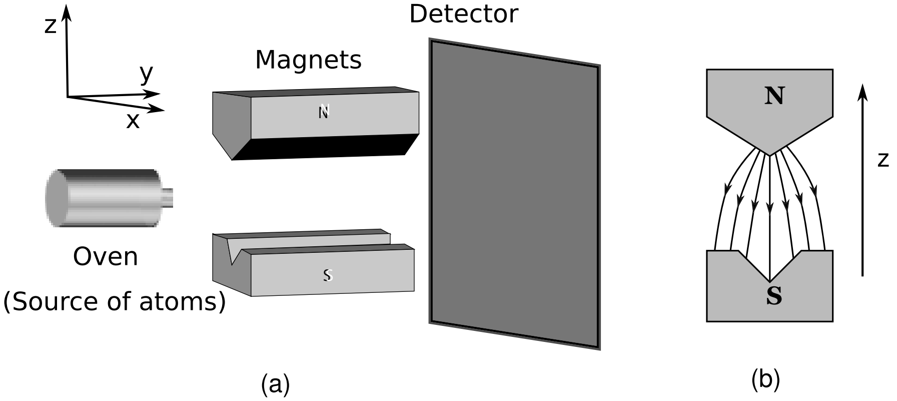


FIGURE 2.1: (a) The Stern–Gerlach Setup. (b) The inhomogeneous magnetic field between asymmetric pole pieces.

The arrangement is such that in the region the electron beam passes through, magnetic field is nearly constant in direction (taken to be $\hat{\boldsymbol{z}}$)[1] but has a strong z-dependent change in magnitude, i.e., $\overrightarrow{\boldsymbol{B}} \approx B(z)\hat{\boldsymbol{z}}$. The force on the dipole when placed in this field is

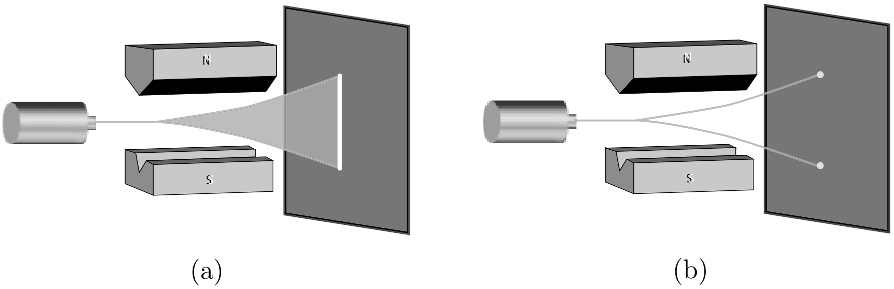
$$\overrightarrow{\boldsymbol{F}} = \overrightarrow{\boldsymbol{\nabla}}(\vec{\boldsymbol{\mu}} \cdot \overrightarrow{\boldsymbol{B}}) = \overrightarrow{\boldsymbol{\nabla}}(\mu_z B(z)) = \mu_z \frac{\partial B(z)}{\partial z}\hat{\boldsymbol{z}}. \tag{2.1}$$

Thus the atom is deflected along the z-axis by an amount proportional to the z-component of its magnetic moment. Remember: since a magnetic field can deflect magnetic moments depending on their magnitudes and directions, it

[1] By convention, in physics experiments, the coordinate system is aligned to the direction of the magnetic field, which is always taken to be the z-axis.

can be used to *select* particular magnetic moments. The net magnetic moment of a collection of atoms is just the vector sum of all the individual atomic magnetic moments. A beam of atoms having a specific constant magnetic moment along a particular direction is said to be **polarized**. It is possible to produce a beam of polarized atoms by specific procedures.

The beam of silver atoms used in the original Stern–Gerlach experiment was produced by heating silver in an oven. Each atom emerges with a random direction of magnetic moment and the net magnetic moment is zero. If such an *unpolarized* beam is sent into the non-uniform magnetic field, then since each atomic magnetic moment is arbitrarily oriented, the z-component of the magnetic moment could vary between $\pm\mu$. So we expect the beam to spread between two extreme limits, which define the value μ of the magnetic moment (see Figure 2.2(a)).



(a)

(b)

FIGURE 2.2: The Stern Gerlach experiment: (a) The classically expected result. (b) What was actually observed.

Suppose, for simplicity, that this experiment is performed with a beam of hydrogen atoms. The hydrogen atom consists of a single electron and a proton. Classically one can think of the electron as orbiting the proton. The atom has associated with it energy that has various contributions. At the first level, the contribution depends on the electrostatic energy, determined by the distance between the positive nucleus and the negative electron, i.e., the orbit radius. Let's assume that this is the minimum possible, or the "ground state" radius. Second, the energy depends on the orbital velocity of the electron, contributing to the orbital angular momentum of the atom. Finally, if the atom is subjected to a magnetic field, its interaction with the field contributes to the energy. The classical analogy however, is severely limited, because radius, velocity, and component of magnetic moment along the magnetic field direction, all can take continuous possible values whereas an atom's energy is *quantized*: takes on only certain discrete values. Correspondingly, the atom is said to exist in possible quantized *energy states*.

The lowest energy state (s-state) has a symmetrical distribution of velocities such that there is no net circulating velocity. Therefore, in this state, the atom is expected to have zero magnetic moment since the average "current" is zero. This means that the beam, when it passes through the Stern–Gerlach setup, will just proceed without deflecting or spreading. The silver atoms used in the original experiment also have zero average magnetic moment. The experiment with hydrogen was also subsequently performed, in 1927 [56].

When the experiment was actually performed, there were *two* surprises. The beam of atoms *did not pass through undeflected.* Nor did it spread, but instead *split into two beams* symmetrically about the central axis, one up and one down. Measuring the positions of the beams indicated a value of $\pm\frac{1}{2}$, in appropriate units,[2] for the magnetic moment! The appearance of Planck's constant h (numerically 6.63×10^{-34}Js), in the magnitude, is a signature of the quantum nature of this property. Whose magnetic moment? The atom in s-state has no net magnetic moment, but has a lone electron. So this had to be an *intrinsic* moment associated with the electron in the atom. Thus, *the magnetic moment of the electron is allowed only to take one of two discrete values!* Classically, the magnetic moment is proportional to the angular momentum of the system. Here, the electron magnetic moment is proportional to a property called intrinsic spin, which mathematically behaves like angular momentum. Thus was discovered the *spin* of the electron, a quantum property that is allowed only two possible values, plus or minus a half.

A word of caution is in place here. In the previous paragraphs we gave a description of the atom in a classical way, to help you form a picture of the physics involved. However, this description is severely limited. In truth, the orbiting of the electrons about the nucleus is not like point particles revolving in space. Nor is the electron really spinning about its axis, it is a point particle with no extension in space! We want to emphasize that the electron spin is a purely quantum mechanical concept, and is physically probed by virtue of its interaction with a non-uniform magnetic field.

You can well imagine that the choice of a particular direction for the magnetic field inhomogeneity cannot affect the value of the magnetic moment of the electron: so even if the apparatus was tilted along any direction, the results would remain the same.

Thus an atom with a single electron, described on the basis of its spin alone, is a 2-state quantum system, well suited as a candidate qubit. We will now illustrate the properties of a qubit using this experiment.

2.2 Quantum State: Basis States

The Stern–Gerlach setup of Figure 2.1 with the direction of inhomogeneity of the magnetic field defined as the z-axis is going to be the basis for defining

[2] The standard unit for atomic magnetic moment is the *Bohr Magneton*, given by $\dfrac{e\hbar}{2m_e c}$, numerically equal to about 5.8×10^{-5}eV/T, where m_e is the mass of the electron, c is the speed of light, and $\hbar = h/2\pi$ is the (reduced) Planck constant.

and measuring electron spin. Let's put it in a box and abbreviate as SG$_z$. The incident beam consists of unpolarized electrons. The machine SG$_z$ produces as output two beams, one above the $z = 0$ axis with electrons of "up" spin and the other below $z = 0$ with electrons of "down" spin. The SG$_z$ machine thus manufactures definite quantum states out of an arbitrary beam. If we isolate or "'ilter out" either of these two states by blocking the other beam, the surviving beam is said to be polarized, and each electron in that beam is in a specific quantum state, called a *basis state.* A quantum state is represented as an angular bracket with a descriptive label inside: a very versatile and useful notation due to Dirac [29]. The two basis states are represented as $|\uparrow\rangle$ and $|\downarrow\rangle$. (Note that these states are defined with respect to a direction of inhomogeneity of an applied magnetic field.) We can thus use the SG$_z$ filter to *prepare* electrons in a predefined quantum state.

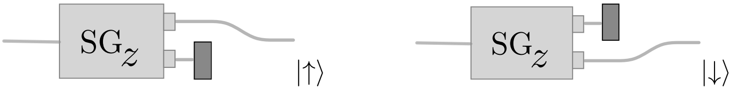


FIGURE 2.3: The SG$_z$ filters (the paths of the beams are bent back to $z = 0$ using suitable magnets).

The two SG$_z$ filters, producing the two basis states, are illustrated in Figure 2.3. (The paths of the beams can be bent back to the $z = 0$ axis by using appropriate magnets.)

We thus not only use the SG$_z$ as a measuring tool for determining the state of an electron, but also as a factory for preparing a known state. This state will be labelled by the spin component along the z-direction.

Suppose a beam of electrons in an unknown state is analyzed using an SG machine. The intensity of a particular output beam can be thought of as the number of electrons in the input beam that are in the corresponding output state. However there are subtleties here. A particular electron in the input beam *randomly* chooses the up or down output port of the machine. From the fraction of the total number of electrons that exit from a particular port, we can deduce the *probability* of the incident electron being in that particular state. This is how quantum mechanics works. We collect a set of statistics of probabilities from measurements and then infer the properties of the system and its state. This is the reason quantum mechanics is often described as a probabilistic theory.

A system could be in a purely quantum mechanical state, with quantum probabilities, and is said to be a *pure* quantum state. However, classical uncertainties could also be present in a given system, in which case the system is said to be in a *mixed* quantum state. For example, the unpolarized beam of silver atoms from the oven in the Stern–Gerlach experiment is actually in a mixed state. We will see more of this distinction in later chapters. For our present introduction, however, we will assume that our systems are always in pure quantum states only.

Let's denote the (unknown but pure) input state of the electrons in the beam by $|i\rangle$ and the output state by $|o\rangle$. The probability of obtaining this particular output is given by

$$\mathcal{P}(|i\rangle \rightarrow |o\rangle) = \frac{\text{no. of electrons in state } |o\rangle}{\text{Total number of electrons in input beam}}. \tag{2.2}$$

We are going to use this information to build a mathematical picture of the spin states of the electron. Let's introduce a notation for this probability (in anticipation of the next chapter). In quantum mechanics, a process such as just described, is associated with a *probability amplitude*, which is in general complex, denoted by $\langle o|i\rangle$. Following the lead of optics, where the intensity of a beam of light is the square of the amplitude of the resultant electric field, the probability of getting output $|o\rangle$ from input $|i\rangle$ is represented by

$$\mathcal{P}(|i\rangle \rightarrow |o\rangle) = |\langle o|i\rangle|^2. \tag{2.3}$$

The reason for amplitudes taking possibly complex values will be seen when we look at the phenomenon of interference between amplitudes in a later section.

The importance of basis states is that when an experiment is performed to measure the state of an electron, the result is invariably one of the basis states. For quantum computation, these basis states represent the bits 0 and 1. They are quantum states and we write them as

$$|0\rangle \equiv |\uparrow\rangle; \quad |1\rangle \equiv |\downarrow\rangle. \tag{2.4}$$

This representation is called the *computational basis*. We can think of these basis states as analogous to the classical bits 0 and 1.

2.2.1 Superpositions

A generic (unknown) state $|\psi\rangle$, before measurement in a particular basis, has the potential to be in either basis state. Suppose the probability amplitudes for measuring the state to be $|\uparrow\rangle$ is a complex number α and that for $|\downarrow\rangle$ is β. We express this fact mathematically by writing $|\psi\rangle$ as a linear superposition

$$|\psi\rangle = \alpha|\uparrow\rangle + \beta|\downarrow\rangle. \tag{2.5}$$

Out of a beam of N electrons, a fraction $|\alpha|^2$ would end up in the upper beam and the fraction $|\beta|^2$ would be in the lower beam. Since all the electrons emerge from the process, the total probability must be one:

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

This unknown state $|\psi\rangle$, with the potential to be in either of the basis states, is said to be in a *superposition* of the basis states.

The peculiarity of a superposition state is that until the system is measured, the state is not definite. It is difficult to visualize, with our classical minds, an object that is in both basis states at once!

2.2.2 Choice of different bases

The magnetic moment, and therefore spin, has three spatial components, and we think of spin as a vector in space with x, y, and z components. To completely determine the spin of an electron, we would need to measure all three components. We could do this by setting up SG machines with magnetic inhomogeneities along the x and y directions as well.

Let's try this on our beam of electrons, whose initial spin state is unknown. We first pass the beam through SG$_z$. Those electrons in the spin state $|\uparrow_z\rangle$ are filtered and sent into an SG$_x$ machine (Figure 2.4). Now classically we would expect that the x spin component should be zero. However, we get two equally intense spin up and spin down beams!

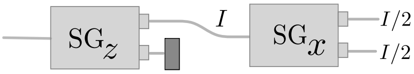


FIGURE 2.4: Measuring S_x after S_z.

This experiment says that an electron in the basis state $|\uparrow_z\rangle$ has definite probabilities of being in both basis states in the x-basis

$$\begin{aligned} \mathcal{P}(|\uparrow_z\rangle \to |\uparrow_x\rangle) &= \frac{1}{2}, \\ \mathcal{P}(|\uparrow_z\rangle \to |\downarrow_x\rangle) &= \frac{1}{2}. \end{aligned} \tag{2.6}$$

Prior to the measurement, the state of the input to the SG$_x$ is actually a superposition of the spin-x basis states:

$$|\uparrow_z\rangle = \frac{1}{\sqrt{2}}|\uparrow_x\rangle + \frac{1}{\sqrt{2}}|\downarrow_x\rangle. \tag{2.7}$$

The coefficients in this superposition are the probability amplitudes, and they are squared to get probabilities.

Now let's block the lower beam in the above experiment. This step constitutes a measurement of the incoming beam and filtering out the spin down components with respect to x. In the third step we pass the $|\uparrow_x\rangle$ beam through an SG$_z$ (Figure 2.5). What is the output?

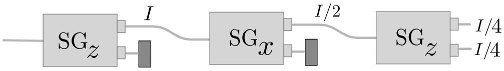


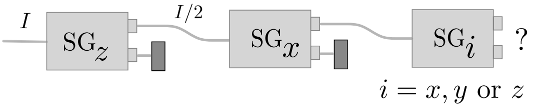
FIGURE 2.5: Successive measurements of S_z, S_x and S_z.

If you expect that only the $|\uparrow_z\rangle$ beam is seen, you are wrong! We once again obtain equally intense spin up and spin down beams.

This brings out a peculiar property of quantum measurements: that the measurement of S_x in the second stage has destroyed all the information about the z component of the spin that the incoming beam had. The third stage sees only the $|\uparrow_x\rangle$ state that is incoming at that stage, which is an equal superposition of $|\uparrow_z\rangle$ and $|\downarrow_z\rangle$. If the experiment is performed to determine S_y instead of S_x on $|\uparrow_z\rangle$ we get the same result: equal components of $|\uparrow_y\rangle$ and $|\downarrow_y\rangle$.

Another feature of quantum mechanics brought out by this experiment is that the three components of the spin vector do not have definite values on the same particle! If we measure one component to know it exactly then the other two become equally indefinite! This is enshrined in a principle known as the "uncertainty principle," more accurately called the *indeterminacy principle.* This principle states that there exist observables that are incompatible with each other with respect to measurement, for example the three components S_x, S_y, and S_z of the spin vector, and accurately determining one of them makes the others indeterminate. We will state this principle in mathematical terms in the next chapter.

Exercise 2.1. What is the output from the series of Stern–Gerlach machines shown below?



2.2.3 Characteristics of basis states

Suppose we have a spin-up beam (the output from SG$_z$ ↑). Feeding it into another SG$_z$ ↑, we would get an output $|\uparrow_z\rangle$ beam of the same intensity (see Figure 2.6).

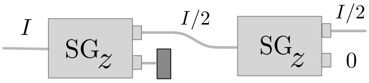


FIGURE 2.6: Repeated spin measurements.

This is represented by the probability amplitude $\langle\uparrow_z|\uparrow_z\rangle = 1$.

If we pass the spin-up beam through an SG$_z$ ↓ filter, then we get no output beam: $\langle\uparrow|\downarrow\rangle = 0$. A similar experiment with SG$_z$ ↓ filter will show us that $\langle\downarrow|\uparrow\rangle = 0$ and $\langle\downarrow|\downarrow\rangle = 1$.

At this stage, let us agree to treat the spin states $|i\rangle$ and $|o\rangle$ as vectors, and think of

$$\langle o|i\rangle$$

as the vector inner product of the states. Also switch to the notation of qubits:

$$|\uparrow_z\rangle \equiv |0\rangle, \quad |\downarrow_z\rangle \equiv |1\rangle.$$

The above properties summarize as

- $\langle 0|1\rangle = 0 \implies$ they are orthogonal states
- $\langle 0|0\rangle = 1 = \langle 1|1\rangle \implies$ they are unit vectors

Thus the set of basis states $\{|0\rangle, |1\rangle\}$ form an orthonormal basis for the state space of the qubit.

2.3 An Experiment to Illustrate Superpositions

An experiment such as that in Figure 2.6 represents a measurement outcome: the average value of the z-component of spin of the input beam is $+1/2$, obtained by the weighted probabilities of spin 1/2 at the up port and spin $-1/2$ at the down port, which in quantum notation is:

$$\langle S_z\rangle_{|i\rangle} = \frac{1}{2}|\langle\uparrow_z|\uparrow_z\rangle|^2 + (-\frac{1}{2})|\langle\downarrow_z|\uparrow_z\rangle|^2 = \frac{1}{2}.$$

Similarly, the experiment in Figure 2.4 gives us the average value of the x-component of spin the input beam $|\uparrow_z\rangle$, which is zero:

$$\langle S_z\rangle_{|i\rangle} = +\frac{1}{2}|\langle\uparrow_x|\uparrow_z\rangle|^2 + (-\frac{1}{2})|\langle\downarrow_x|\uparrow_z\rangle|^2 = 0.$$

Consider a beam of spin-up electrons from the filter SG$_z$ ↑. We now set up a second Stern–Gerlach machine, but rotated by an angle θ to z. Let us label this machine "SG$_\theta$". Into this machine we pass the beam of $|\uparrow_z\rangle$ electrons. What would be the measured output? The schematic setup is in Figure (2.7) and the outputs are $|\uparrow_\theta\rangle$ state at the up port and $|\downarrow_\theta\rangle$ state at the down port. We wish to predict the probabilities of each state.

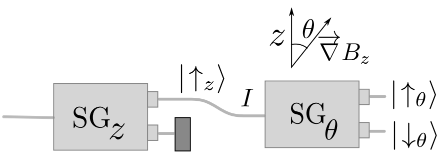


FIGURE 2.7: An experiment with the SG along an arbitrary direction θ.

Now the "classical" projection $\cos\theta \times 1/2$ of the spin of the incoming beam along the θ direction, gives us an *average value* for the measured spin, weighing in both the output ports. The intensity of the up beam gives an average spin of $+1/2$ with probability $\mathcal{P}(\uparrow_\theta) = |\langle\uparrow_\theta|\uparrow_z\rangle|^2$ and $-1/2$ with probability $\mathcal{P}(\downarrow_\theta) = |\langle\downarrow_\theta|\uparrow_z\rangle|^2$. The total probability of this happening is $\cos\theta$. So we have

$$\frac{1}{2}\cos\theta = \frac{1}{2}\mathcal{P}(\uparrow_\theta) - \frac{1}{2}\mathcal{P}(\downarrow_\theta). \tag{2.8}$$

We also have an equation for conservation of probability:

$$1 = \mathcal{P}(\uparrow_\theta) + \mathcal{P}(\downarrow_\theta). \tag{2.9}$$

From these two, we get the probabilities of the up and down spin states in the output:

$$\mathcal{P}(\uparrow_\theta) = \cos^2\frac{\theta}{2}, \qquad \mathcal{P}(\downarrow_\theta) = \sin^2\frac{\theta}{2}. \tag{2.10}$$

Let's check this result by comparing with the special cases:

- $\theta = 0$: SG$_\theta$ = SG$_z$,
$$\mathcal{P}(\uparrow) = 1, \qquad \mathcal{P}(\downarrow) = 0.$$
- $\theta = \pi$: SG$_\theta$ = SG$_{-z}$ or an SG$_z$ turned upside down,
$$\mathcal{P}(\uparrow) = 0, \qquad \mathcal{P}(\downarrow) = 1.$$
- $\theta = \pi/2$: SG$_x$ (or SG$_y$)
$$\mathcal{P}(\uparrow) = \frac{1}{2} = \mathcal{P}(\downarrow)$$

Mathematically we can write the state of the input electron as a superposition of the output states of SG$_\theta$:

$$|{\uparrow_z}\rangle = \alpha|{\uparrow_\theta}\rangle + \beta|{\downarrow_\theta}\rangle,$$

where

$$|\alpha|^2 = \mathcal{P}(\uparrow_\theta) = \cos^2\frac{\theta}{2}, \qquad |\beta|^2 = \mathcal{P}(\downarrow_\theta) = \sin^2\frac{\theta}{2}.$$

The experiment only tells us the magnitudes of the complex amplitudes α and β.

2.4 Interference and Complex Amplitudes

Consider a sequence of Stern–Gerlach tests where an unpolarized beam passes through an SG$_z$ and subsequently through an SG$_x$. Clearly there are four possible outcomes: a screen placed at the end will show up 4 spots as in Figure 2.8.

If we place another SG$_z$ at the end, we expect eight possible outcomes as in Figure 2.9.

If we now slowly turn off the middle SG$_x$, we expect the $x-$separation of the beams to slowly reduce until they coalesce and we expect four outcomes

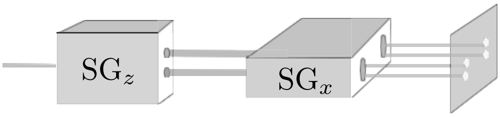


FIGURE 2.8: Successive Stern–Gerlachs in z and x.

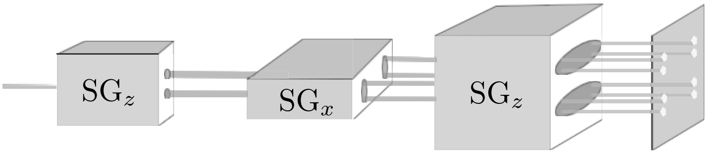


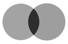
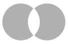
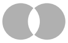
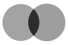
FIGURE 2.9: Successive Stern–Gerlachs z, x and z again.

all vertically separated. However, this is finally equivalent to two SG$_z$'s one after the other, and the result should be just *two* beams vertically separated!

What's wrong here?

Clearly it is the second scenario that should be experimentally observed, and indeed is. The conflict between two pictures can be resolved by considering *interference* between the beams. The final spot intensity at a point on the screen is obtained by the square of the superposition of probability amplitudes of the beams overlapping at that point. The amplitudes for the middle two beams have to be of opposite signature (or *phase*, to use an optics analogy) so that when the x-field is gradually reduced, and the beams merge, they cancel each other when they overlap. The picture for the beams as this happens is schematically indicated in Table 2.1

TABLE 2.1: Interference in the Stern–Gerlach setup.

Beam Amplitudes		Spot intensity on screen			
Left	Right	as SG$_x$ is slowly turned off			
		L	R	L R	Final (SG$_x$ off)
+	+	●	●		●
+	−	●	●		no spot
−	+	●	●		no spot
−	−	●	●		●

In the language of optics, we can say that the middle two beams, having opposite phases, destructively interfere so that the corresponding intensity is zero.

This was an example of interference between just two states. In more complex situations where multiple states exist, each state must be associated with a phase that is in general complex. It is this phenomenon of interference in quantum mechanics that calls for description of states with complex amplitudes. In mathematical language, each state is associated with a complex vector, one that has a magnitude as well as a phase.

The Stern–Gerlach setup we have described in this chapter serves multiple purposes for us. First, it demonstrates the quantum property of spin of an electron as a prototypical two-state system that can be used as a qubit. Second, we can use the setup to prepare a quantum system in a predefined state: initializing it to $|0\rangle$ or $|1\rangle$ by filtering out one of the outputs. Third, the setup can be used as a detector to measure the state of the input beam.

Exercise 2.2. Suppose that *one* of the four beams output from the middle SG$_x$ were blocked (in Figure 2.9). What would be the intensities of the various output beams?

Box 2.1: Polarization States of Light

The quantum spin described in this chapter is novel and has no classical analog. However, the same picture of a 2-dimensional Hilbert space emerges from considering the polarization states of light. This example is worth considering, as it will be particularly useful later when we use light for quantum information processing. The analogy with spin is also complete, with a classical picture to peg our understanding on.

Classically, light is electromagnetic radiation, with oscillating electric and magnetic fields. The form of the fields comes from solutions to Maxwell's equations. It is easier to detect the electric field, so we will describe light by its electric field vector. The important parameters that describe a monochromatic light wave are its wavelength λ and angular frequency (color) ω and the wave vector $\vec{\boldsymbol{k}} = \frac{2\pi}{\lambda}\hat{\boldsymbol{k}}$ giving the direction of propagation. The direction $\hat{\boldsymbol{k}}$ is conventionally taken to be $\hat{\boldsymbol{z}}$, just as SG$_z$ is the standard for the spin system. An important property of the electromagnetic wave in free space is transversality: the electric field vector always lies in a plane perpendicular to $\hat{\boldsymbol{k}}$. The direction of the electric field vector is known as its *polarization.* This direction could be constant, as in linearly polarized light, or rotate in the polarization plane, as in circularly polarized light.

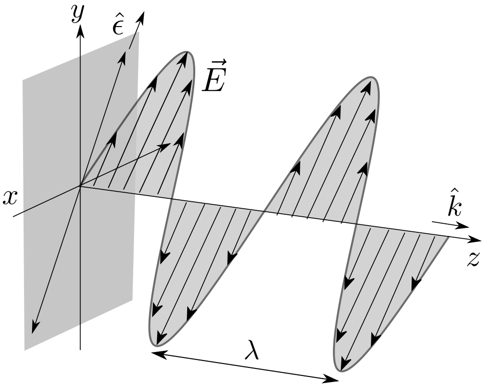


FIGURE 2.10: Linearly polarized light wave.

Let's first look at linearly polarized light with the $\overrightarrow{\boldsymbol{E}}$ field oscillating along a direction we'll call $\hat{\boldsymbol{\epsilon}}$ (Figure 2.10). It is possible to produce such light by passing unpolarized light from a monochromatic source through a *polaroid filter*. Such a filter has a "pass axis" that allows polarizations parallel to this axis alone to be transmitted through. The field is described by

$$\overrightarrow{\boldsymbol{E}} \quad = \quad E_0 \hat{\boldsymbol{\epsilon}} \cos(kz - \omega t). \tag{2.11}$$

The intensity of light is given by the magnitude square of the electric field. If a second polarizer, with its pass-axis at an angle δ to the first, is placed in the light path, then only the component of the $\overrightarrow{\boldsymbol{E}}$ field along this angle is passed through. So the electric field of the transmitted light is $E \cos \delta$ in a direction parallel to the new pass axis. The intensity of the light falls by a factor $\cos^2 \delta$ (Figure 2.11).

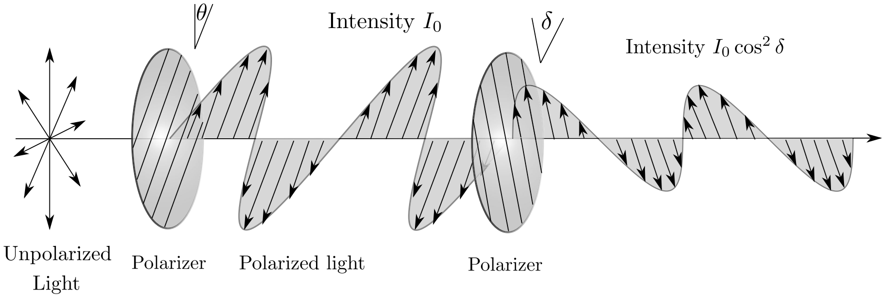


FIGURE 2.11: Effect of a linear polarizer on unpolarized light; subsequent polarizer allows only a component $\propto \cos^2 \delta$ through.

More generally, the electric field could have components oscillating along the

$\hat{\boldsymbol{x}}$ and the $\hat{\boldsymbol{y}}$ directions with different amplitudes and even different phases:

$$\overrightarrow{\boldsymbol{E}} = E_1\hat{\boldsymbol{x}}e^{i(kz-\omega t)} + E_2\hat{\boldsymbol{y}}e^{i(kz-\omega t+\phi)}. \tag{2.12}$$

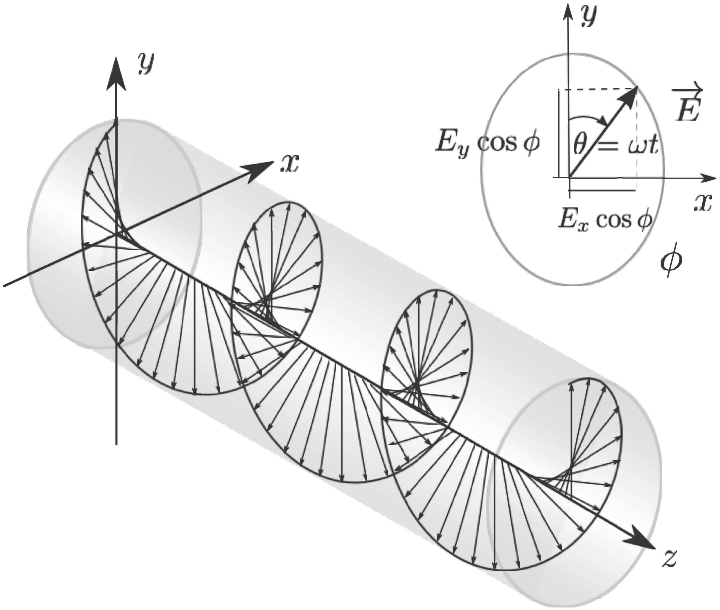


FIGURE 2.12: Light with elliptic polarization: the electric field vector traces out an ellipse in the x-y plane

One can define the polarization vector

$$\hat{\boldsymbol{\epsilon}} = \frac{E_1}{|\overrightarrow{\boldsymbol{E}}|}\hat{\boldsymbol{x}} + \frac{E_2}{|\overrightarrow{\boldsymbol{E}}|}e^{i\phi}\hat{\boldsymbol{y}}.$$

This is in general elliptical polarization (Figure 2.12). The special case $\phi = \pi/2$ corresponds to circular polarization while $\phi = 0$ is linear polarization.

It is easy to see that if $\hat{\boldsymbol{x}}$-polarized light is incident on a $\hat{\boldsymbol{y}}$-polarizer (a polaroid filter with its pass axis along the $\hat{\boldsymbol{y}}$ direction), no light passes through. We can thus define two orthogonal polarization states of light corresponding to the vertical ($\hat{\boldsymbol{y}}$) and horizontal ($\hat{\boldsymbol{x}}$) directions. This experiment is analogous to the Stern–Gerlach-z machine, with up and down ports being analogous to the vertical and horizontal polarizations. We can thus draw analogy between the vector space of light polarizations and the spin Hilbert space:

$$\begin{aligned} |0\rangle &\leftrightarrow \updownarrow \\ |1\rangle &\leftrightarrow \leftrightarrow . \end{aligned}$$

To create the optical analogue of the states produced by SG$_x$ filters, we will need to use polarizers that are rotated by 45° with respect to the earlier ones. Light produced by these polarizers can be in polarization states ⤢ and ⤡ defined by the 45° orthogonal directions:

$$
\begin{aligned}
\text{⤢} &= \frac{1}{\sqrt{2}}(\hat{\boldsymbol{x}} + \hat{\boldsymbol{y}}) \\
\text{⤡} &= \frac{1}{\sqrt{2}}(\hat{\boldsymbol{x}} - \hat{\boldsymbol{y}}).
\end{aligned}
$$

It is easy to see that if ⤢ light is incident on an $\hat{\boldsymbol{x}}$ or $\hat{\boldsymbol{y}}$ polarizer then 50% of the incident beam passes through. Similarly for ⤡.

The analogy of the SG$_y$ basis is with right and left circular polarization. From Equation 2.12, we see that light with electric field rotating in the plane of polarization arises due to a phase difference of $\pi/2$ between the x- and y-components. The complex notation is most suitable for expressing this phase relationship (using $i = e^{i\pi/2}$):

$$
|\uparrow_y\rangle \quad \leftrightarrow \quad \circlearrowleft = \frac{1}{\sqrt{2}}(\hat{\boldsymbol{x}} + i\hat{\boldsymbol{y}}) \tag{2.13}
$$

$$
|\downarrow_y\rangle \quad \leftrightarrow \quad \circlearrowright = \frac{1}{\sqrt{2}}(\hat{\boldsymbol{x}} - i\hat{\boldsymbol{y}}) \tag{2.14}
$$

The necessity of complex probability amplitudes becomes clear now, due to considerations of phase being unavoidable.

Part II

Theoretical Framework

Chapter 3

The Essentials of Quantum Mechanics

How do we describe the state of a quantum mechanical physical system in a mathematically precise way? How do we ascribe physical properties to this system? How does the system evolve in time? How do its properties change when it interacts with another system or a force? How do we measure and determine its properties? These are the questions whose answers are encoded in the laws of quantum mechanics to be set down in this chapter. They have been inscribed in this form after nearly half a century of experimental observations, theoretical modeling, and intellectual gymnastics to tie up the two in a satisfactory and robust structure.

3.1 The State Space

When you describe the state of a physical system, you collect all the parameters required to fully specify it: for instance, the state of a ball may be specified by its position in space, its velocity, and maybe its rate of spin; the state of a volume of gas by its temperature and its pressure. If you are trying to describe a quantum system like a hydrogen atom, you may think specifying the position and velocity of the atom and its constituents, the nucleus and the electron would give the quantum state. Whether this is true, or even possible in principle, depends on how you are trying to see the atom: which properties you are trying to measure and what experiments you are using to measure its properties,

So we first identify a system, an *isolated* set of physical properties that we have experimental access to and are trying to describe. The quantum state of the system, denoted by the notation $|\text{state}\rangle$,[1] is represented by measured values

[1] The notation due to Dirac that we use in quantum mechanics may need some more clarification. A state is labelled abstractly as $|\psi\rangle$, or as $|0\rangle$, or as $|x_i\rangle$. The labels are just mnemonics to tag the state. They may be numbers but are not the components of the vector in any basis. For instance, $|0\rangle$ does not mean the zero vector, for which we will use the notation $\vec{\mathbf{0}}$. The 0 used as a label is an indication of a first basis vector in the computational basis.

of the physical properties used to describe it. It is important to know which properties are independent of each other, measuring which do not interfere with the other properties. The outcome of the measurement could be one of many possibilities. Each possibility labels a different state. The set of all these forms the **state space** of the system.

In the last chapter, the particular property of spin of an electron was targeted for study, by designing the Stern–Gerlach experiment. This led to a state space of two states.

The properties of a quantum state turn out to conform to those of a *vector* in the mathematical sense: a member of a *complex vector space* (see Box 3.1), with a notion of *norm* or *inner product* defined on it. Such a vector space is called a *Hilbert space* (see Box 3.2). The vector describing a state must also have unit norm, since we will be attaching a notion of probabilities to the state. The complex vector may also have imaginary components, but an overall phase factor is unimportant since we have no means of measuring it. Thus a state is unit vector in complex space, modulo an overall phase factor.

Postulate 1. *The state of an isolated quantum mechanical system is a unit vector in Hilbert space.*

Box 3.1: Linear Vector Space

A **vector space** $\mathcal{V}$ is a set of objects $\mathbf{v}$, called **vectors**, that abstractly satisfy the properties of closure under an operation of addition, and under multiplication by a scalar which belongs to a field $\mathcal{F}$, that for example could be real or complex numbers. In what follows, a vector is designated by a boldface, such as $\mathbf{v}$, while a scalar is not.

The axioms defining a vector space are

1. **Addition**: one can define an operation "+" such that for any vectors $\mathbf{v_i} \in \mathcal{V}^n$,

   (A1) $\mathcal{V}^n$ is **closed** under +: $\mathbf{v_1} + \mathbf{v_2} \in \mathcal{V}^n$,

   (A2) + is **commutative**: $\mathbf{v_1} + \mathbf{v_2} = \mathbf{v_2} + \mathbf{v_1}$,

   (A3) + is **associative**: $(\mathbf{v_1} + \mathbf{v_2}) + \mathbf{v_3} = \mathbf{v_1} + (\mathbf{v_2} + \mathbf{v_3})$.

   (A4) $\exists$ a **zero vector** or additive identity $\mathbf{0} \in \mathcal{V}^n$ such that $\mathbf{v} + \mathbf{0} = \mathbf{v}$.

   (A5) For each $\mathbf{v} \in \mathcal{V}$, $\exists$ an additive inverse $-\mathbf{v} \in \mathcal{V}^{\mathbf{n}}$ such that $\mathbf{v}+(-\mathbf{v}) = \mathbf{0}$.

2. **Scalar Multiplication**: for any scalar $\alpha \in \mathcal{F}$ and vector $\mathbf{v_i} \in \mathcal{V}^n$,

   (M1) $\mathcal{V}$ is closed under scalar multiplication: $\alpha\mathbf{v} \in \mathcal{V}^n$,

   (M2) For the multiplicative identity 1, we have $1\mathbf{v} = \mathbf{v}$,

   (M3) Multiplication by the scalar 0 gives the zero vector: $0\mathbf{v} = \mathbf{0}$, (M4) Associativity: $\alpha(\beta\mathbf{v}) = (\alpha\beta)\mathbf{v}$,

(M5) Distributivity over vector addition: $\alpha(\mathbf{v_1} + \mathbf{v_2}) = \alpha\mathbf{v_1} + \alpha\mathbf{v_2}$,

(M6) Distributivity over scalar addition: $(\alpha + \beta)\mathbf{v} = \alpha\mathbf{v} + \beta\mathbf{v}$

The element $-\mathbf{v} = (-1)\mathbf{v}$ is the additive inverse of $\mathbf{v}$.

Vectors can be represented by *components* if we choose a set of "coordinates" or **basis** vectors for the representation. A **basis** for a vector space consists of a set of vectors $\{\mathbf{e_i}\}$ whose defining properties are:

1. they are linearly independent: no basis vector can be expressed as a linear combination of the other basis vectors; **no** set of numbers $\{a_i\}$ can be found such that
$$\sum_i a_i\mathbf{e}_i = \mathbf{0}.$$

2. they span the vector space $\mathcal{V}$: any vector $\mathbf{v} \in \mathcal{V}$ can be expressed as a linear combination of the basis vectors:
$$\mathbf{v} = c_1\mathbf{e}_1 + c_2\mathbf{e}_2 + \cdots + c_n\mathbf{e}_n.$$

The index i counts the basis vectors: $i = 1...n$. The total number n of basis vectors is the **dimension** of the vector space. This dimension can be finite or infinite. The index i can be discrete or continuous. We will only be dealing here with finite-dimensional complex vector spaces.

A vector space in general has more than one basis. A vector represented by its components is also represented as a column matrix:

$$\mathbf{v} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix}.$$

To save space, we will also represent this as the transpose of a row vector

$$\mathbf{v} = \begin{bmatrix} c_1 & c_2 & \dots c_n \end{bmatrix}^T.$$

This representation is extremely useful when we consider transformations of a vector space into another by linear maps, which can be represented by matrices.

Vector spaces are familiar to us from 3-dimensional spacial vectors, but the above definitions generalize such properties to a larger class of objects. We find that even continuous functions of complex numbers that are infinitely differentiable and vanish fast at infinity form a vector space.

3.1.1 Basis states

We saw in the previous chapter how to describe the *spin* state of an electron.[2] The "system" in this case is just that property of an electron that responds to a gradient in an applied magnetic field. The state of this system is a member of a 2-dimensional vector space. This is because this spin can take one of only two possible values, $\pm\hbar/2$. An electron in either of these states is described by the *basis vectors*

$$|0\rangle = |{+\tfrac{\hbar}{2}}\rangle, \quad |1\rangle = |{-\tfrac{\hbar}{2}}\rangle.$$

A general state $|\psi\rangle$ is a linear combination of these basis vectors with complex coefficients:

$$|\psi\rangle = \alpha_0|0\rangle + \alpha_1|1\rangle.$$

As we will show, these basis states are mutually **orthogonal** and are **normalized**, so that they form an **orthonormal** basis. This is similar to representing a physical 2-dimensional vector in terms of its components along two orthogonal directions. This vector is represented as the column matrix of its components: $[\ \alpha_0 \quad \alpha_1\ ]^T$.

We can easily generalize this to higher dimensions. Such a picture is relevant when the set of basis states for the system is larger. For example, the system may be the magnetic moment of a spin-$\frac{3}{2}$ atomic nucleus. This object would have four possible states distinguished in a non-uniform magnetic field:

$$\{|j\rangle\} = \{|\tfrac{3}{2}\rangle, |\tfrac{1}{2}\rangle, |{-\tfrac{1}{2}}\rangle, |{-\tfrac{3}{2}}\rangle\}.$$

Another example is the electronic energy of the hydrogen atom. This system actually has a countable infinity of possible energy states labelled by the so-called "principal quantum number" n:

$$\{|n\rangle\}, \quad n = 0, 1, 2 \ldots.$$

This Hilbert space is actually infinite dimensional, though we might say the dimensionality is "countable." If we were concentrating on the position states of a particle confined to a line then the possible states are a continuous infinity labelled by the values of the position x:

$$\{|x\rangle\}, \quad -\infty \leq x \leq +\infty.$$

This Hilbert space is also infinite dimensional, and the dimensionality is continuous and uncountable.

[2] The spin space is a *subspace* of the total state space of an electron, which contains descriptors of all possible compatible measurable properties of the electron. This Hilbert space can be expressed as a *direct product* of the independent subspaces.

Box 3.2: Hilbert Space

The linear vector space of Box 3.1 turns into something rich enough to represent states of a physical system if a little more structure is added to it. We now have a Hilbert space $\mathcal{H}^n$ which is defined as a complex vector space with an **inner product** $(.,.) \in \mathbb{C}$ which satisfies

(I1) $(\mathbf{v}, \mathbf{v}) \geq 0, \quad (\mathbf{v}, \mathbf{v}) = 0 \quad \text{iff} \quad \mathbf{v} = \mathbf{0}.$

(I2) $(\mathbf{u}, \mathbf{v}) = (\mathbf{v}, \mathbf{u})^*$

(I3) $(\mathbf{u}, \alpha\mathbf{v}) = \alpha(\mathbf{u}, \mathbf{v})$

(I4) $(\mathbf{v_1} + \mathbf{v_2}, \mathbf{v_3}) = (\mathbf{v_1}, \mathbf{v_3}) + (\mathbf{v_2}, \mathbf{v_3}).$

With this structure in place, a vector space becomes a **pre-Hilbert space**, and is a Hilbert space if the dimension is finite. For infinite-dimensional Hilbert spaces, one needs the additional criterion of the space being **complete** under the inner product, which we will not discuss here.

3.1.2 Inner product

In order to be able to define orthogonality and the "size" of a vector, we need the notion of an *inner product.* This is just like the *dot product* of two vectors. This is basically a rule for assigning a (complex) number to a pair of vectors.

For this we define a dual vector space $\mathcal{V}^\dagger$ of same dimensions. Vectors in this space are represented by row matrices $[\alpha_0 \ \alpha_1 \ ... \ \alpha_n]$. The dual of the vector $|v\rangle = [v_1 \ v_2 \ \ldots \ v_n]^T$ is represented by $\langle v| = [v_1^* \ v_2^* \ ... \ v_n^*]$ where the * denotes complex conjugation. Thus the matrix representation of the dual vector $\langle v|$ is the *complex conjugate transpose* of $|v\rangle$, denoted by $|v\rangle^\dagger$.

The inner product of vectors $|\phi\rangle$ and $|\psi\rangle$ is defined as the complex number $\langle\phi|\psi\rangle$. (This bracket $\langle\cdot|\cdot\rangle$ for inner product is the origin of the Dirac **bra-ket** notation: the **ket** vector $|\cdot\rangle$ has a dual **bra** vector $\langle\cdot|$ and their product gives the "bra(c)ket".)

If $|\psi\rangle = [\alpha_1 \ \alpha_2 \ \ldots \ \alpha_n]^T$ and $|\phi\rangle = [\beta_1 \ \beta_2 \ \ldots \ \beta_n]^T$ then their inner product is

$$\langle\phi|\psi\rangle = \beta_1^*\alpha_1 + \beta_2^*\alpha_2 + ... + \beta_n^*\alpha_n. \tag{3.1}$$

Some of the consequences of this definition are:

- Norm of a vector is defined as $\|v\| = \sqrt{\langle v|v\rangle}$. A vector is said to be normalized if it has unit norm. An arbitrary vector can be normalized by dividing it by its norm:

$$|v\rangle_{\text{norm}} = \frac{|v\rangle}{\|v\|}.$$

- Orthogonality: vectors $|\psi\rangle$ and $|\phi\rangle$ are orthogonal to each other iff $\langle\psi|\phi\rangle = 0$.

A Hilbert space spanned by n linearly independent vectors $|i\rangle, i = 1...n$ is said to be n-dimensional. The set then defines a *basis* for the space. A generic vector in the space can be expressed as a linear combination of these basis states with complex coefficients c_i:

$$|\psi\rangle = \sum_{i=1}^{n} c_i |i\rangle. \tag{3.2}$$

In matrix representation, the *natural* basis vectors are the n-column vectors

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}; |1\rangle = \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix}; \quad ... \quad |n\rangle = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}.$$

It is useful to note that the label i is also the integer representation of the binary number represented by the string of components of the i^{th} basis vector! An example for $n = 2$ gives us the basis for qubits:

$$\begin{bmatrix} c_0 \\ c_1 \end{bmatrix} = c_0 \begin{bmatrix} 1 \\ 0 \end{bmatrix} + c_1 \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

Experimentally, the orthonormal basis states for a quantum system would be states such as a $|\uparrow\rangle$ and $|\downarrow\rangle$, that are mutually exclusive results of direct measurements of a physical property. An arbitrary state would be a linear combination of these basis states. The complex coefficients c_i of the combination are interpreted as **probability amplitudes** for the state to be in the corresponding basis state $|i\rangle$. These probability amplitudes can also be expressed as

$$c_i = \langle i|\psi\rangle.$$

So we can write the general quantum state as

$$|\psi\rangle = \sum_{i=1}^{n} \langle i|\psi\rangle \; |i\rangle.$$

Since probabilities must add to 1, we get what is called the **normalization** condition

$$\sum_{i=1}^{n} |c_i|^2 = 1.$$

This is the same as the condition $\langle\psi|\psi\rangle = 1$, i.e., the state $|\psi\rangle$ must be normalized, or be a unit vector in Hilbert space. A state vector that does not have unit norm can be *normalized* by dividing it by its norm.

The first axiom of quantum mechanics is a statement that embodies all this.

Exercise 3.1. *Normalize the vectors* $\begin{bmatrix} 1 \\ 1 \end{bmatrix}$ *and* $\begin{bmatrix} 1 \\ -1 \end{bmatrix}$. *Show that they are orthogonal.*

Exercise 3.2. *Normalize the state* $|0\rangle - 2i|1\rangle$.

3.1.2.1 Meaning of inner product

Inner product of vectors gives the component of one vector in the direction of the other. Similarly for quantum states, the inner product $\langle\psi|\phi\rangle$ is the probability amplitude that one state is along the other. For example,

$$|\psi\rangle = \frac{1}{\sqrt{3}}|0\rangle + \sqrt{\frac{2}{3}}|1\rangle.$$

Then the inner product

$$\langle 0|\psi\rangle = \frac{1}{\sqrt{3}}$$

is the probability amplitude that the state $|\psi\rangle$ has spin up.

As another example, the Hilbert space of position states of a particle along the x-axis would have an infinite set of basis states $|x\rangle$. A general state $|\psi\rangle$ has a probability amplitude $\langle x|\psi\rangle = \psi(x)$ of being found at the location x. This probability amplitude as a function of position is better known as the **wave function** of the particle.

The inner product also comes in when describing the outcome of a process that transforms a system from initial state $|\psi_i\rangle$ to final state $|\psi_f\rangle$. The mod-square of the probability amplitude for this process is then the probability that such an event can occur:

$$\mathcal{P}(|\psi_i\rangle \to |\psi_f\rangle) \quad = \quad |\langle\psi_f|\psi_i\rangle|^2. \tag{3.3}$$

This statement, one of the underpinnings of quantum mechanics, is known as the **Born rule** after Max Born,[3] who first postulated it.

3.1.3 Phases

The coefficients in the expansion of a state in terms of the basis states are complex numbers in general. We saw one reason for this in the last chapter:

[3] In a 1926 paper in a German journal, Born mentioned the probability interpretation in a footnote.

we need to account for interference when probability amplitudes are added. Now a complex number has a modulus and a phase: $z = x + iy$ has magnitude $r = \sqrt{x^2 + y^2}$ and a phase $\phi = \tan^{-1} y/x$. r and ϕ are real numbers and we express the same complex number in modular form as $z = re^{i\phi}$. Suppose we write

$$|\psi\rangle = r_1 e^{i\theta_1}|0\rangle + r_2 e^{i\theta_2}|1\rangle. \tag{3.4}$$

Different values of $r_1\theta_1$ and $r_2\theta_2$ give different vectors. For a given vector $|\psi\rangle$, we can factor out one of the phases to write

$$|\psi\rangle = e^{i\theta_1}\Big(r_1|0\rangle + r_2 e^{i(\theta_2-\theta_1)}|1\rangle\Big) \tag{3.5}$$

The factored phase θ_1 is called a **global phase**. This cannot be measured by any experiment since experiments only measure probabilities. In other words the above state is experimentally indistinguishable from the state

$$|\psi'\rangle = r_1|0\rangle + r_2 e^{i(\theta_2-\theta_1)}|1\rangle,$$

since $|\langle\psi|\psi'\rangle|^2 = 1$. What is measurable, however, is the **relative phase** $(\theta_2 - \theta_1)$, which will show up in an interference experiment. The set of all states differing by a global phase is called a **ray** in Hilbert space.

Thus the space of quantum states of a system is the space of rays in Hilbert space, also called the **projective Hilbert space**. We will not emphasize this difference in what follows, but it is a point to be kept in mind.

The fact that relative phases between components in a superposition state are very important will become more relevant when we consider operations on quantum systems that impart selective phases to one basis state, say $|1\rangle$. For instance, consider an operation

$$|0\rangle \to |0\rangle; \quad |1\rangle \to e^{i\phi}|1\rangle.$$

Though such an operation produces indistinguishable states out of basis states, the effect will be non-trivial on superposition states, since it would introduce a relative phase between the $|0\rangle$ and $|1\rangle$ components:

$$|\psi\rangle = c_1|0\rangle + c_2|1\rangle \to c_1|0\rangle + e^{i\phi}c_2|1\rangle.$$

3.2 Observables

The state space may be said to be defined by its basis states. How do we identify the basis? We have said that when we measure a physical quantity, the state corresponding to the value measured is a basis state for the system. This brings us directly to the question: which physical quantity shall we choose to measure? Well, the choice is entirely ours. But a certain amount of scientific acumen is necessary to identify the relevant one! In any case, all the physically measurable properties of the system are important: these are called **observables**.

Measurement of a particular observable $\mathcal{O}$ yields a set of possible values, in suitable units, that the observable could take. This set of real numbers characterizes the observable. In mathematical language, these numbers are regarded as the characteristic values or **eigenvalues** of an *operator* representing the observable. The set of characteristic values is called the **spectrum** of the observable. This is indeed a full specification of the observable. But in quantum mechanics, we try to attach the notion of an operator to the observable. What is an operator?

3.2.1 Operators

An operator $\hat{O}$, formally, is a method for transforming a vector $|v\rangle$ into another, $|v'\rangle$. Expressed mathematically, the operator acts from the left of the vector:

$$\hat{O}|v\rangle = |v'\rangle.$$

In the language of linear algebra, operators are represented as matrices. For example:

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

Box 3.3: Diagonalizable Operators and the Spectral Theorem

An operator $\hat{\mathcal{A}}$ is said to satisfy an **eigenvalue equation** if there exist some vectors $|\epsilon_i\rangle$ that are transformed into multiples of themselves:

$$\hat{\mathcal{A}}|\epsilon_i\rangle = a_i|\epsilon_i\rangle.$$

The number a_i is called an *eigenvalue* corresponding to the vector $|\epsilon_i\rangle$, which is called an **eigenvector**. The eigenvalues may be distinct (simple) or some of them may be equal (multiple). In the latter case they are said to be **degenerate**. Not all matrices satisfy eigenvalue equations. Those that do are called **diagonalizable**. This name is due to the **spectral theorem** which says that such operators can be expressed as diagonal matrices in the basis of their eigenvectors with the eigenvalues as the diagonal elements:

$$\hat{\mathcal{N}} = \sum_i n_i|\epsilon_i\rangle\langle\epsilon_i|.$$

This statement essentially means that for a diagonalizable matrix, we can change basis to one in which the matrix is diagonal. In other words, there exists a non-singular matrix $\hat{S}$ bringing $\hat{\mathcal{A}}$ to diagonal form $\hat{\mathcal{N}}$ by a similarity transformation:

$$\hat{S}\hat{\mathcal{A}}\hat{S}^{-1} = \hat{\mathcal{N}}.$$

A special class of diagonalizable operators is important in quantum mechanics, those that commute with their adjoint:

$$\hat{\mathcal{N}}^{\dagger}\hat{\mathcal{N}} = \hat{\mathcal{N}}\hat{\mathcal{N}}^{\dagger}.$$

Such an operator is called a **normal** operator. Some kinds of normal operators especially relevant to us are:

1. Unitary operators: $\hat{\mathcal{U}}^{\dagger} = \hat{\mathcal{U}}^{-1}$
2. Hermitian operators: $\hat{\mathcal{H}}^{\dagger} = \hat{\mathcal{H}}$

   (Also anti-Hermitian operators: $\hat{\mathcal{A}}^{\dagger} = -\hat{\mathcal{A}}$
3. Positive operators: $\hat{P} = \hat{\mathcal{M}}\hat{\mathcal{M}}^{\dagger}$. These operators are also Hermitian.

The following important properties of eigenvalues are to be noted

1. A Hermitian operator has real eigenvalues.
2. A positive operator has positive eigenvalues.
3. A unitary operator has eigenvalues of unit modulus, i.e., of the form $e^{i\theta}$ for real θ.

Suppose we find the dual of the transformed vector $|v'\rangle$. What is the operator in dual space that would take $\langle v|$ to $\langle v'|$? The answer is the adjoint operator $\hat{O}^{\dagger}$ ('O-dagger'), defined by the equation

$$\langle v|\hat{O}^{\dagger} = \langle v'|.$$

We can also compare the transformed and original vectors by their inner product with another vector $|w\rangle$, and thus define the adjoint by

$$(|w\rangle, \hat{O}|v\rangle) = (\hat{O}^{\dagger}|w\rangle, |v\rangle). \tag{3.6}$$

We can see that each side of Equation 3.6 is equivalent to $\langle w|\hat{O}|v\rangle = \langle w|v'\rangle = \langle w'|v\rangle$, where $\langle w'| = \langle w|\hat{O}^{\dagger}$. Notice that the action of the adjoint is from the right.

The matrix representation of $\hat{O}^{\dagger}$ is the complex conjugate transpose of the matrix for $\hat{O}$. Thus the dual vector $\langle v|$ is sometimes also called the adjoint of the ket vector $|v\rangle$.

3.2.2 Self-adjoint operators

An operator is said to be self-adjoint if it satisfies

$$\langle v|\hat{A}|w\rangle = \langle v|\hat{A}^\dagger|w\rangle. \tag{3.7}$$

The corresponding matrix is said to be **Hermitian**. An important consequence of self-adjointness is that the eigenvalues will turn out to be real. A self-adjoint operator is thus a good candidate for a physical observable whose values are always real.

Postulate 2. Observables *An observable A in quantum mechanics is usually represented by a self-adjoint operator*[4] *$\hat{A}$. Measurement of A in an experiment gives a real number value α, which is one of the eigenvalues of the operator $\hat{A}$.*

By "measurement of an observable" we mean the setting up of a suitable experiment and determining the value associated with that physical property. We will discuss measurements in quantum mechanics in more detail soon.

For example, the machine SG_z of the previous chapter measures the z-component of the spin, S_z, and yields two possible values $\pm\hbar/2$. The operator corresponding to this spin observable, denoted by $\hat{S}_z$, has eigenvalues $\pm\hbar/2$ and corresponding eigenstates $|0\rangle$ and $|1\rangle$. This means it satisfies the eigenvalue equations

$$\hat{S}_z|0\rangle = \frac{\hbar}{2}|0\rangle, \quad \hat{S}_z|1\rangle = -\frac{\hbar}{2}|1\rangle.$$

Applying the spectral theorem (3.2.1), the matrix representation of $\hat{S}_z$ in the computational basis is:

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \hat{S}_z = \frac{\hbar}{2}\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \tag{3.8}$$

Exercise 3.1. Show that $\hat{S}_z$ is Hermitian.

Exercise 3.2. Solve the eigenvalue equation for $\hat{S}_z$ and show that its eigenvalues are $\pm\hbar/2$.

3.2.3 Basis transformation

We have been saying that the choice of basis depends on the observable we choose to measure. The Hilbert space must be spanned by the bases corresponding to the eigenstates of other observables too. This implies a relationship between different bases for a given system.

[4] We need operators with real eigenvalues. In recent times, non-Hermitian operators also seem to be relevant under certain special conditions, but these need not concern us here.

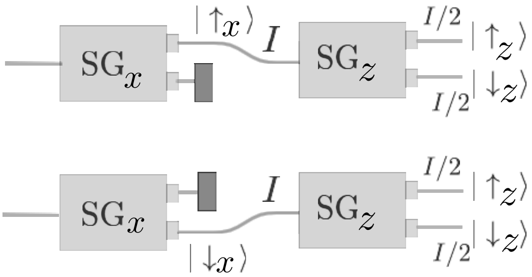

SG$_x$
$|\uparrow_x\rangle$
I
SG$_z$
$I/2$ $|\uparrow_z\rangle$
$I/2$ $|\downarrow_z\rangle$
SG$_x$
I
$|\downarrow_x\rangle$
SG$_z$
$I/2$ $|\uparrow_z\rangle$
$I/2$ $|\downarrow_z\rangle$

FIGURE 3.1: Experiment for determining the eigenstates of $\hat{S}_x$ in the computational basis

Consider the spin observable, related to the magnetic moment, which is a vector in 3-dimensional space. The vector spin observable $\overrightarrow{\boldsymbol{S}}$ has the nature of angular momentum, and has three spatial components: S_x, S_y, and S_z. The machines for measuring these observables would be, respectively, SG$_x$, SG$_y$, and SG$_z$, each with its B field inhomogeneity at right angles to that of the other. But measurement of each of these would give one of two values, $\pm\hbar/2$. This means that in each basis of representation, the eigenstates and the matrix for the operator is given by Equation 3.8.

We would like to represent each of these observables and their eigenstates in the common computational basis $\{|0\rangle, |1\rangle\}$. This, by convention, is the basis of eigenstates of the operator $\hat{S}_z$ which we had written as $|\uparrow_z\rangle$ and $|\downarrow_z\rangle$. What, for instance, is the form of the eigenstates $|\uparrow_x\rangle$ and $|\downarrow_x\rangle$ of $\hat{S}_x$ in this basis? Look at the Stern–Gerlach experiments shown in Figure 3.1.

This says that $|\uparrow_x\rangle$ and $|\downarrow_x\rangle$ are 50-50 superpositions of $|0\rangle$ and $|1\rangle$.

$$|\uparrow_x\rangle = \alpha|0\rangle + \beta|1\rangle, \quad \text{where } |\alpha|^2 = |\beta|^2 = \frac{1}{2}.$$

A similar equation can be written for $|\downarrow_x\rangle$. In fact a similar equation would hold for the eigenstates $|\uparrow_y\rangle$ and $|\downarrow_y\rangle$ of $\hat{S}_y$. Each would need to have different complex coefficients α and β to distinguish them. We can fix these coefficients up to a relative phase: each has magnitude $\frac{1}{\sqrt{2}}$ and some phase which is not fixed experimentally. (See Section 2.3.) By convention, we choose the relative phase angle ϕ to be zero for $|\uparrow_x\rangle$ and π for $|\uparrow_y\rangle$ and fix the rest by demanding orthogonality.

Example 3.2.1. Basis transformation from $\hat{S}_z$ to $\hat{S}_x$ basis: to emphasize that $|\uparrow_x\rangle$ and $|\downarrow_x\rangle$ are also a different set of basis vectors, let us denote them by $|0_x\rangle$ and $|1_x\rangle$. Experiment is consistent with

$$|0_x\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right).$$

We also require $\langle 0_x|1_x\rangle = 0$, which is consistent with

$$|1_x\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right).$$

It is also easy to see that the basis vector transformation can be written in matrix form as

$$\begin{pmatrix} |0_x\rangle \\ |1_x\rangle \end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}\begin{pmatrix} |0\rangle \\ |1\rangle \end{pmatrix}$$

Henceforth, we will switch to a less cumbersome notation for the spin operators. We consider the following dimensionless operators, each having eigenvalues ±1 and the same eigenstates as those of corresponding spin operators.

$$\hat{X} = \frac{2}{\hbar}\hat{S}_x; \qquad \text{eigentates } |0_x\rangle, |1_x\rangle, \tag{3.9a}$$

$$\hat{Y} = \frac{2}{\hbar}\hat{S}_y; \qquad \text{eigentates } |0_y\rangle, |1_y\rangle, \tag{3.9b}$$

$$\hat{Z} = \frac{2}{\hbar}\hat{S}_z; \qquad \text{eigentates } |0_z\rangle \equiv |0\rangle, |1_z\rangle \equiv |1\rangle. \tag{3.9c}$$

Box 3.4: Basis Transformations among the $\hat{X}$, $\hat{Y}$ and $\hat{Z}$ Bases

$$|0_x\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right) \tag{3.10a}$$

$$|1_x\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right) \tag{3.10b}$$

$$|0_y\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + i|1\rangle\right) \tag{3.10c}$$

$$|1_y\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle - i|1\rangle\right] \tag{3.10d}$$

Exercise 3.3. Verify from these definitions that $\{|0_x\rangle, |1_x\rangle\}$ are an orthonormal set. Similarly for $\{|0_y\rangle, |1_y\rangle\}$.

Exercise 3.4. Express $\{|0_y\rangle, |1_y\rangle\}$ in terms of $\{|0_x\rangle, |1_x\rangle\}$.

It is important to realize that a change of basis is effected by a linear transformation: When a basis $\{|i\rangle\} \to \{|j\rangle\}$ then for each $|j\rangle$ we can find a set of n complex coefficients U_{ij} such that

$$|j\rangle = \sum_i U_{ij}|i\rangle. \tag{3.11}$$

These components U_{ij} can be shown to form the components of a unitary matrix U. The change of basis can be visualized as a sort of rotation of the axes that span the Hilbert space.

Example 3.2.2. Unitarity of the transformation matrix for basis change: from Equation 3.11, let us use the orthogonality of the basis $\{|j\rangle\}$ to write

$$\begin{aligned}
\langle j'|j\rangle = \sum_{i'} U^*_{j'i'}\langle i'|\sum_i U_{ij}|i\rangle &= \delta_{j'j} \\
\Longrightarrow \sum_{i'}\sum_i U^*_{j'i'}U_{ij}\langle i'|i\rangle &= \delta_{j'j} \\
\Longrightarrow \sum_i U^*_{j'i}U_{ij} &= \delta_{j'j}
\end{aligned}$$

But this last equation is exactly the condition $U^\dagger U = \mathbb{1}$ for unitarity of U.

3.2.4 Outer product representation for operators

From the components of two vectors, we can construct a matrix by the **outer product**. For vectors $|v_1\rangle = [a_1a_2...a_n]^T$ and $|v_2\rangle = [b_1b_2...b_n]^T$, this is denoted by $|v_1\rangle\langle v_2|$ and represented by a matrix given by

$$|v_1\rangle\langle v_2| = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} \begin{bmatrix} b_1^* & b_2^* & ... & b_n^* \end{bmatrix} = \begin{bmatrix} a_1b_1^* & a_1b_2^* & ... & a_1b_n^* \\ a_2b_1^* & a_2b_2^* & ... & a_2b_n^* \\ \vdots & \vdots & ... & \vdots \\ a_nb_1^* & a_nb_2^* & ... & a_nb_n^* \end{bmatrix} \tag{3.12}$$

Operators on a Hilbert space can be represented in terms of outer products of the basis vectors of the space: a matrix A with matrix elements A_{ij} is the expansion

$$A \equiv \sum_{i,j} A_{ij}|i\rangle\langle j|.$$

Conversely, in the above basis, a matrix A has elements

$$A_{ij} = \langle i|A|j\rangle,$$

where i is the row index and j is the column index. The space of matrices is thus a linear vector space with basis "vectors" given by the matrices $|i\rangle\langle j|$ composed of the outer products of the basis vectors of the Hilbert space. For example, in 2 dimensions, the basis matrices will be

$$\begin{aligned}
|0\rangle\langle 0| &= \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \\
|0\rangle\langle 1| &= \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \\
|1\rangle\langle 0| &= \begin{bmatrix} 0 \\ 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, \\
|1\rangle\langle 1| &= \begin{bmatrix} 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}.
\end{aligned} \tag{3.13}$$

A 2×2 matrix is represented as

$$\begin{aligned}
A &= A_{00}|0\rangle\langle 0| + A_{01}|0\rangle\langle 1| + A_{10}|1\rangle\langle 0| + A_{11}|1\rangle\langle 1| \\
&= A_{00}\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + A_{01}\begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} + A_{10}\begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} + A_{11}\begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \\
&= \begin{bmatrix} A_{00} & A_{01} \\ A_{10} & A_{11} \end{bmatrix}.
\end{aligned}$$

The spectral theorem can then be expressed in the form

$$\hat{\mathcal{A}} = \sum_i a_i |a_i\rangle\langle a_i| \tag{3.14}$$

where the a_i are the eigenvalues of $\hat{\mathcal{A}}$ corresponding to its eigenvectors $|a_i\rangle$. Note how we use the eigenvalue itself as the label for the corresponding eigenstate! The set of eigenvalues is called the **spectrum** of the operator and this equation is called the **spectral resolution** of the operator.

Example 3.2.3. Matrix representation for the spin operators in the computational basis:$\{|0\rangle, |1\rangle\} = \{|\uparrow\rangle_z, |\downarrow\rangle_z\}$:

$$\hat{S}_z = \frac{\hbar}{2}|0\rangle\langle 0| - \frac{\hbar}{2}|1\rangle\langle 1| = \frac{\hbar}{2}\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}.$$

To construct the representation for $\hat{S}_x$ in this basis we first note that it is diagonal in the basis of its own eigenvectors:

$$\hat{S}_x = \frac{\hbar}{2}|\uparrow\rangle_x\langle\uparrow|_x - \frac{\hbar}{2}|\downarrow\rangle_x\langle\downarrow|_x$$

We can now use the basis transformation equations 3.10 and write

$$\begin{aligned}
\hat{S}_x &= \frac{\hbar}{2}\left\{\frac{(|0\rangle+|1\rangle)}{\sqrt{2}}\frac{(\langle 0|+\langle 1|)}{\sqrt{2}}+\frac{(|0\rangle-|1\rangle)}{\sqrt{2}}\frac{(\langle 0|-\langle 1|)}{\sqrt{2}}\right\} \\
&= \frac{\hbar}{2}\left\{|0\rangle\langle 1|+|1\rangle\langle 0|\right\} \\
&= \frac{\hbar}{2}\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}.
\end{aligned}$$

To specify an operator, one has to give its action on the computational basis states. This will give the matrix representation of the operator in the natural basis.

Exercise 3.5. Show that in the computational basis,

$$\hat{S}_y = \frac{\hbar}{2}\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}.$$

Exercise 3.6. Show by diagonalizing these representations of $\hat{S}_x$ and $\hat{S}_y$, you get the eigenvectors of Equations 3.10.

Exercise 3.7. Calculate the effect of $\hat{S}_x$ and $\hat{S}_y$ on the states $|0\rangle$and $|1\rangle$.

3.2.5 Functions of operators

Very often in dealing with the mathematics of quantum mechanics we encounter the need to evaluate functions of operators, the very simplest being powers. That is easily dealt with since operators can be composed by multiplication. Functions that can be expressed in power series can then be computed in terms of the powers of the operators. For example,

$$\begin{aligned}
\cos\hat{A} &= \mathbb{1}-\frac{1}{2!}\hat{A}^2+\frac{1}{4!}\hat{A}^4+\cdots \\
\sin\hat{A} &= \hat{A}-\frac{1}{3!}\hat{A}^3+\frac{1}{5!}\hat{A}^5+\cdots \\
\exp\hat{A} &= \mathbb{1}+\hat{A}+\frac{1}{2!}\hat{A}^2+\frac{1}{3!}\hat{A}^3+\cdots
\end{aligned}$$

More complicated functions can also be dealt with, if we use the spectral decomposition, Equation 3.14, of the operator. If $\hat{A}$ has eigenvalues a_i and corresponding eigenvectors $|a_i\rangle$, we have

$$\hat{A} = \sum_i a_i|a_i\rangle\langle a_i|$$

$$f(\hat{A}) \quad = \quad \sum_i f(a_i)|a_i\rangle\langle a_i|. \tag{3.15}$$

Thus, to find the function $f(\hat{A})$ when $\hat{A}$ is not already in diagonal form, you must first find the similarity transformation S that diagonalizes $\hat{A}$:

$$S^{-1}\hat{A}S \quad = \quad \sum_i a_i|a_i\rangle\langle a_i|.$$

S can be constructed out of the eigenvectors of $\hat{A}$: they are the columns of S.

The Dirac Notation

We pause to remark on the extreme versatility of the $|.\rangle$ notation of Dirac: various combinations that are meaningful and unambiguous are summarized below for your reference.

TABLE 3.1: The Dirac notation and its properties.

Notation	Class	Name, property, and meaning
$\|a\rangle$	$\in \mathcal{H}$	Vector or *ket* .
$\langle a\|$	$\in \mathcal{H}^\dagger$	Dual vector $\|a\rangle^\dagger$ or bra
$\langle b\|a\rangle$	$\in \mathbb{C}$	Inner product $\langle b\| \cdot \|a\rangle$, a complex number
$\|a\rangle\langle b\|$	an operator on $\mathcal{H}$	Outer product
$\|a\rangle\|b\rangle \equiv \|a\rangle \otimes \|b\rangle$	$\in \mathcal{H}_1 \otimes \mathcal{H}_2$ for $\|a\rangle \in \mathcal{H}_1$ and $\|b\rangle \in \mathcal{H}_2$	Direct product

Thus a sequence of objects in the Dirac notation occurring in any expression will give one unambiguous interpretation, with the operations of inner, outer, or direct products implicit. For example, if a ket vector follows a bra vector as in $\langle a||b\rangle$ the only way to interpret the result is as an inner product $\langle a|b\rangle$. As another example, sequence $\langle a|b\rangle\langle c|d\rangle$ may be interpreted as two inner products $\langle a|b\rangle$ times $\langle c|d\rangle$ or as the matrix element $\langle a|\hat{B}|d\rangle$ of the matrix $\hat{B} = |b\rangle\langle d|$ but the result is the same (complex) number. You will see more illustrations of this versatility in worked examples throughout the book. You must train yourself to recognize and utilize this feature to the fullest.

Box 3.5: The Pauli Spin Matrices

The spin observables are multiples by $\hbar/2$ of the following matrices:

$$\sigma_1 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}; \sigma_2 = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}; \sigma_3 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

The identity matrix $\mathbb{1} \equiv \sigma_0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ can also be included in the set $\{\sigma_i\}$. These are of prime importance to us. They have interesting properties:

1. $\hat{S}_i = \frac{\hbar}{2}\sigma_i$ where $i = 1, 2, 3$ stand for x, y and z.
2. $\sigma_i^2 = \mathbb{1} \implies \sigma_i^\dagger = \sigma_i = \sigma_i^{-1}$

   So they are Hermitian as well as unitary and therefore have eigenvalues ± 1.
3. Any unitary 2×2 matrix (operator) can be expressed as a linear combination of the σ_is.

In the context of quantum computation, the matrices σ_1, σ_2, and σ_3 are denoted as X, Y, and Z (Equations 3.9). Their eigenvectors corresponding to the eigenvalues ± 1 are sometimes also denoted as $|X\pm\rangle, |Y\pm\rangle$ and $|Z\pm\rangle$, respectively.

3.3 Measurement

We mentioned measurement in the context of observables, but measurement has a very important role in quantum mechanics. It is important to identify what we mean by measurement. Measurement is basically an experimental procedure meant to determine the value of a physical observable. The procedure must be carefully designed so as not to alter the value of the observable being measured. A cartoon of the process of measurement is given in Figure 3.2. Regardless of specific procedures used for any particular observable, the process of ideal measurement and the result of measurement are axiomatized as follows:

Postulate 3. Measurement

1. *Measurement of an observable $\hat{A}$ of a system in the state ψ yields an eigenvalue a of the observable, corresponding to an eigenstate $|a\rangle$, with*

the probability

$$\mathcal{P}(a) = |\langle a|\psi\rangle|^2. \tag{3.16}$$

2. *Measurement causes the state of the system to collapse to the eigenstate $|a\rangle$:*

$$|\psi\rangle \xrightarrow{\textit{Measure } \hat{A}\textit{, obtain } a} |a\rangle. \tag{3.17}$$

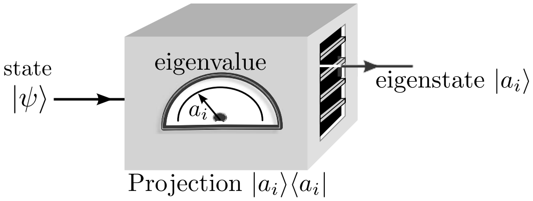


FIGURE 3.2: The measurement process for measuring an observable A with values a. Note that this is distinct from an operator A acting on $|\psi\rangle$ to transform it to $A|\psi\rangle$.

To describe this process by an operator acting on the state, we introduce the **projection operator**: its action is to project the state along another state $|a\rangle$:

$$\hat{\mathbb{P}}_a = |a\rangle\langle a|. \tag{3.18}$$

When this acts on a general state, it *projects* the component of that state along the vector $|a\rangle$ (Figure 3.3). If we express $|\psi\rangle$ in the basis $\{|i\rangle\}$, we have

$$\begin{aligned} |\psi\rangle &= \sum_i c_i|i\rangle, \\ \hat{\mathbb{P}}_j|\psi\rangle &= |j\rangle\langle j|\psi\rangle = c_j|j\rangle. \end{aligned}$$

Box 3.6: Properties of Projection Operators

1. A projection operator is *idempotent*: $\hat{\mathbb{P}}_i^2 = \hat{\mathbb{P}}_i$.

2. A projection operator divides the Hilbert space $\mathcal{H}$ into orthogonal subspaces $\mathcal{U}$ and $\mathcal{V}$:

$$\forall|x\rangle \in \mathcal{H}, \hat{\mathbb{P}}|x\rangle = |u\rangle \in \mathcal{U}, (\mathbb{1} - \hat{\mathbb{P}})|x\rangle = |v\rangle \in \mathcal{V}, \langle v|u\rangle = 0.$$

This is because an n-dimensional Hilbert space has n orthogonal directions and associated projectors. This also translates into

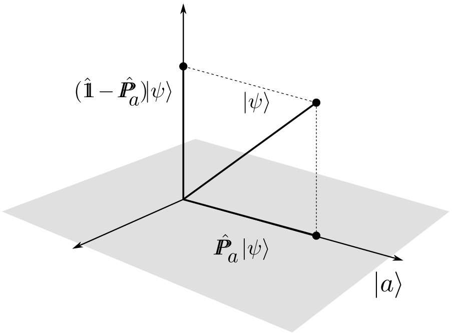

$(\hat{\mathbb{1}} - \hat{\boldsymbol{P}}_a)|\psi\rangle$
$|\psi\rangle$
$\hat{\boldsymbol{P}}_a|\psi\rangle$
$|a\rangle$

FIGURE 3.3: Projection operator $|a\rangle\langle a|$ acting on a state.

(a) the projectors are mutually orthogonal:

$$\hat{\mathbb{P}}_i\hat{\mathbb{P}}_j = \delta_{ij}\hat{\mathbb{P}}_j. \tag{3.19}$$

(b) The projectors are complete:

$$\sum_i \hat{\mathbb{P}}_i = \sum_i |i\rangle\langle i| = \mathbb{1}. \tag{3.20}$$

This completeness relation is also called the **resolution of identity** and is usefully employed in many proofs.

3. Projection operators are Hermitian: $\hat{\mathbb{P}}_i^\dagger = \hat{\mathbb{P}}_i$. Orthogonality of projection operators is also sometimes expressed as

$$\hat{\mathbb{P}}_i^\dagger\hat{\mathbb{P}}_i = \hat{\mathbb{P}}_i. \tag{3.21}$$

Measuring the observable $\hat{A}$ and obtaining a value a is a probabilistic event. The probability amplitude of obtaining this value is the coefficient of expansion of ψ in the basis $\{a\}$, which is (see Equation 3.2) $\langle a|\psi\rangle$. This can also be written as

$$\mathcal{P}(a) = |\langle a|\psi\rangle|^2 = \langle\psi|\hat{\mathbb{P}}_a|\psi\rangle.$$

This is the meaning of the first measurement postulate.

Measuring an operator $\hat{\mathcal{A}}$ in the state $|\psi\rangle$ forces that state into one of the eigenstates $|a\rangle$ of the operator. This is actually one kind of ideal measurement, called a **projective measurement**, equivalent to the operation

$$|\psi\rangle \xrightarrow{\text{Measure A, obtain } a} \hat{\mathbb{P}}_a|\psi\rangle = |a\rangle\langle a|\psi\rangle.$$

Note that this process is *not unitary* . The final state is not normalized. If we can capture this process by some **measurement operator** $\hat{\mathcal{M}}_A$ acting on the state, then we must have

$$\hat{\mathcal{M}}_A|\psi\rangle = \frac{|a\rangle\langle a|\psi\rangle}{|\langle a|\psi\rangle|^2}.$$

But such an operator cannot be defined independent of $|\psi\rangle$. Measurements must thus have a separate status in quantum mechanics, and the process is not represented by a unitary operator. This is the meaning of the second measurement postulate.

What is the value of an observable in a quantum state?

Someone gives you an electron and asks you: what is the spin? How will you answer? If you measure S_x, S_y, or S_z you will get one of two answers, at random. Any observable you measure gives one of its eigenvalues at random. The state has probabilistic information about each eigenvalue. The meaning of this is statistical: (i) take a very large number of identical copies of the state $|\psi\rangle$: a statistical *ensemble*, (ii) perform the measurement of $\hat{\mathcal{A}}$ on each copy, then if you expand ψ in the basis of eigenvectors a of $\hat{\mathcal{A}}$,

$$|\psi\rangle = \sum_a c_a|a\rangle$$

then a fraction $|c_a|^2$ times you will obtain a as the result of the measurement. The *value* of the physical observable $\mathcal{A}$ is the statistical average value obtained by all these measurements. This is called the **expectation value** of the operator $\hat{\mathcal{A}}$ in the state $|\psi\rangle$ denoted by

$$\begin{aligned}\langle\hat{\mathcal{A}}\rangle_\psi &= \sum_a \mathcal{P}(a)a = \sum_a |c_a|^2 a \\ &= \sum_a \langle\psi|a\rangle\langle a|\psi\rangle a = \sum_a |\psi\rangle\hat{\mathcal{A}}|a\rangle\langle a|\psi\rangle \\ &= \langle\psi|\hat{\mathcal{A}}|\psi\rangle,\end{aligned}$$

where in the last step we have used the resolution of the identity.

We can thus define statistically the mean value of an observable. The statistics of measurement is incomplete without the notion of the variance about the mean. We define the variance $\Delta^2\mathcal{A}$ as

$$\Delta^2\mathcal{A} = \langle\mathcal{A}^2\rangle - \langle\mathcal{A}\rangle^2$$

The square root of the variance, the standard deviation, is called the *error* or **uncertainty** in the value of $\mathcal{A}$.

Example 3.3.1. Expectation value of $\hat{S}_z$ and $\hat{S}_x$ in the state $|0\rangle$:

$$\begin{aligned}
\langle \hat{S}_z \rangle_0 &= \langle 0|\hat{S}_z|0\rangle = \langle 0|\frac{\hbar}{2}|0\rangle \\
&= \frac{\hbar}{2} \\
\langle \hat{S}_x \rangle_0 &= \langle 0|\hat{S}_x|0\rangle = \langle 0|\frac{\hbar}{2}|1\rangle \\
&= 0.
\end{aligned}$$

Box 3.7: The Uncertainty Principle

This principle is one of the foundation pillars of quantum mechanics, first enunciated by Werner Heisenberg. More accurately called the **indeterminacy principle**, this states that some physical observables are "incompatible" with each other, in the sense that on measurement in a given state, it is not possible to get sharp values of both. In fact, the uncertainty in one observable is inversely related to that in the other. Classic examples are position and momentum, and also the three components of the spin vector.

Mathematically, compatibility is related to the commutation of the operators: whether the order of operation of two operators matters or not. For two operators $\hat{A}$ and $\hat{B}$ the commutator is defined as the operator expressing this difference in ordering:

$$\hat{C} = [\hat{A}, \hat{B}] \equiv \hat{A}\hat{B} - \hat{B}\hat{A}.$$

It can be shown that the product of uncertainties of two operators measured in a state $|\psi\rangle$ is related to their commutator:

$$\left(\Delta\hat{A}\Delta\hat{B}\right)_\psi \geq \frac{1}{2}\left\langle[\hat{A}, \hat{B}]\right\rangle_\psi. \tag{3.22}$$

Note that experimentally uncertainty refers to the standard deviation from the mean of a statistically large set of measurements of the observable, made on identically prepared states. Physically the meaning of the uncertainty principle is that if we perform a set of measurements of observable A and B in an ensemble prepared in a state $|\psi\rangle$, then the products of the uncertainties of the two observables is limited by the expression on the right, related to their commutator. Experimental uncertainties would add to this limit. Thus in principle, the uncertainty in either of a pair of observables that do not commute can never be zero.

3.4 Evolution

An isolated system is said to evolve when its state changes with time. The change in state would take place by the action of an operator on it. This action cannot take it out of the Hilbert space, and must preserve its norm. Therefore the evolution operator $\hat{\mathcal{U}}$ has to satisfy some conditions.

$$\begin{aligned}
|\psi\rangle \xrightarrow{\hat{\mathcal{U}}} |\psi'\rangle &= \hat{\mathcal{U}}|\psi\rangle \\
\langle\psi'|\psi'\rangle &= \langle\psi|\hat{\mathcal{U}}^\dagger\hat{\mathcal{U}}|\psi\rangle \\
\text{If } \langle\psi'|\psi'\rangle &= \langle\psi|\psi\rangle \\
\text{then } \hat{\mathcal{U}}^\dagger &= \hat{\mathcal{U}}^{-1}
\end{aligned}$$

Such an operator is called unitary:

$$\hat{\mathcal{U}}^\dagger\hat{\mathcal{U}} = \mathbb{1}.$$

In quantum computation, any operation we wish to perform on a qubit must be represented by such an operator. One of the important consequences of this is that since any unitary operator is invertible, any quantum operation is reversible.

For example, the Pauli spin operators are unitary and are valid evolution operators. The operation $|0\rangle \rightarrow |1\rangle$and $|1\rangle \rightarrow |0\rangle$is achieved by the σ_1 operator. This operation flips the bits 0 and 1, and is therefore also called the NOT operator X.

Thus evolution is another application of unitary operators in quantum mechanics. The first one we encountered of course was while implementing basis change.

3.4.1 Continuous time evolution

From the physical viewpoint, evolution in time occurs due to interaction of the system with an external force. A characteristic of this "force" is the energy the system has in its presence. This energy is represented by a function called the **Hamiltonian** function H. In a given situation it has to be determined experimentally. The quantum version of the Hamiltonian is the Hamiltonian operator $\hat{H}$. This operator, being an observable, must be Hermitian. Now it turns out that when the Hamiltonian acts on a state vector, it creates an infinitesimal time evolution. This gives a differential version of the time evolution postulate of which there are two (experimentally equivalent) viewpoints or "pictures":

3.4.1.1 Schrödinger viewpoint

Postulate 4. *The evolution in time of a quantum state vector $|\psi(t)\rangle$ is given by the Schrödinger equation:*

$$i\hbar\frac{d|\psi\rangle}{dt} = \hat{H}|\psi\rangle. \tag{3.23}$$

We can try to understand what this implies by formally integrating this equation to solve for $|\psi(t)\rangle$ from $|\psi(t_0)\rangle$. Assuming that the Hamiltonian function is itself explicitly independent of time, we would get

$$|\psi(t)\rangle = \exp\left[-\frac{i}{\hbar}\hat{H}(t-t_0)\right]|\psi(t_0)\rangle.$$

So the unitary operator for time evolution is just

$$\hat{\mathcal{U}}(t_0, t) \equiv \exp\left[-\frac{i}{\hbar}\hat{H}(t-t_0)\right]. \tag{3.24}$$

We can set $t_0 = 0$ and write

$$\hat{\mathcal{U}}(t) = e^{-i\hat{H}t/\hbar}.$$

Here, the exponential of the operator $\hat{H}$ is understood as the infinite sum of powers of $\hat{H}$:

$$e^{-i\hat{H}t/\hbar} \equiv \mathbb{1} - \frac{it}{\hbar}\hat{H} + \frac{1}{2}\frac{it}{\hbar}^2\hat{H}^2 + \cdots,$$

itself an operator that can be expressed as a matrix. You can verify that since $\hat{H}$ is Hermitian, $\hat{\mathcal{U}}(t)$ is indeed unitary.

3.4.1.2 Heisenberg viewpoint

One can focus on the observables being measured instead of the state in which they are measured, and think of evolution as affecting the observable (operator) instead of the state vector. In this picture, the evolution of an observable $\hat{A}(t)$ is given by

$$\begin{aligned} \hat{A}(t) &= \hat{\mathcal{U}}(t)\hat{A}(0)\hat{\mathcal{U}}^\dagger(t) && (3.25)\\ \Longrightarrow \frac{d\hat{A}}{dt} &= \frac{d}{dt}\hat{\mathcal{U}}\hat{A}(0)\hat{\mathcal{U}}^\dagger + \hat{\mathcal{U}}\hat{A}(0)\frac{d}{dt}\hat{\mathcal{U}}^\dagger \\ &= \frac{i}{\hbar}\left(-\hat{H}\hat{\mathcal{U}}\hat{A}(0)\hat{\mathcal{U}}^\dagger + \hat{\mathcal{U}}\hat{A}(0)\hat{H}\hat{\mathcal{U}}^\dagger\right) \\ \frac{d\hat{A}}{dt} &= \frac{i}{\hbar}[\hat{A}(t), \hat{H}], && (3.26) \end{aligned}$$

where the square brackets indicate the commutator $AH - HA$. Here we have assumed that the observable A itself has no explicit time-dependence; that is, t does not occur in its form. If it did then we would have to add the partial derivative of $\hat{A}(t)$ with respect to t. It is straightforward to see that both pictures give the same value for the experimentally observed quantities: the expectation values of observables.

3.5 Composite Systems

We would in general consider not just a single quantum system, representing one qubit, but a multiple qubit system that will consist of distinct and non-interacting component single-qubit systems. The quantum states of the composite system are elements of a larger Hilbert space composed of the single qubit Hilbert spaces. For this we take what is called a **direct product** of the single-qubit basis states, to form basis states of the larger Hilbert space. This direct product is also called a **tensor product**, represented by the symbol $\otimes$. The elements of the tensor product basis consist of ordered sequences of elements from the bases of each of the component Hilbert spaces.

Postulate 5. *The Hilbert space of a composite system $\mathcal{S}$ is the direct product of Hilbert spaces of the components $A, B, C....$:*

$$\mathcal{H}_{\mathcal{S}} = \mathcal{H}_A \otimes \mathcal{H}_B \otimes \mathcal{H}_C... \tag{3.27}$$

If the subsystems have basis states $\{|e_A\rangle\}, \{|e_B\rangle\}...$, then each basis state of the full system is a tensor product of the form

$$|e_i\rangle = |e_A\rangle_i \otimes |e_B\rangle_i \otimes ...$$

A general state of the composite system can be expressed as a linear combination of basis states of the composite Hilbert space.

For example, a 2-qubit system would consist of two non-interacting single qubits (say the individual z-spins of two isolated electrons), each with a 2-d Hilbert space $\mathcal{H}^2$. The Hilbert space of the 2-qubit system is then

$$\mathcal{H}^4 = \mathcal{H}^2 \otimes \mathcal{H}^2. \tag{3.28}$$

If we label the bases of the $\mathcal{H}^2$s as $\{|0\rangle_A, |1\rangle_A\}$ and $\{|0\rangle_B, |1\rangle_B\}$, we get the basis for $\mathcal{H}^4$ as the ordered pairs

$$\begin{aligned}&\{|0\rangle_A, |1\rangle_A\} \otimes \{|0\rangle_B, |1\rangle_B\}\\ &= \{|0\rangle_A \otimes |0\rangle_B, |0\rangle_A \otimes |1\rangle_B, |1\rangle_A \otimes |0\rangle_B, |1\rangle_A \otimes |1\rangle_B\}.\end{aligned} \tag{3.29}$$

The notation $|a\rangle \otimes |b\rangle$ is shortened to $|ab\rangle$ and we write the basis for $\mathcal{H}^4$ as

$$\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}. \tag{3.30}$$

We see binary representations of the numbers 0 to 3 emerging in this 2-qubit system.

In matrix notation, these basis vectors are generated by direct products. To write the direct product of two matrices, we should realize that every element of one matrix is associated with every element of the other. This is done in the following manner:

Definition 3.1. *The tensor product of two matrices A of dimensions $m \times n$ and B of any dimensions is given by*

$$A \otimes B = \begin{bmatrix} A_{11}B & A_{12}B & \dots & A_{1n}B \\ A_{21}B & A_{22}B & \dots & A_{2n}B \\ \vdots & \vdots & & \vdots \\ A_{m1}B & A_{m2}B & \dots & A_{mn}B \end{bmatrix}. \tag{3.31}$$

Thus we have

$$\begin{aligned} |00\rangle = \begin{bmatrix}1\\0\end{bmatrix} \otimes \begin{bmatrix}1\\0\end{bmatrix} = \begin{bmatrix}1\\0\\0\\0\end{bmatrix}; \quad |01\rangle = \begin{bmatrix}1\\0\end{bmatrix} \otimes \begin{bmatrix}0\\1\end{bmatrix} = \begin{bmatrix}0\\1\\0\\0\end{bmatrix}; \\ |10\rangle = \begin{bmatrix}0\\1\end{bmatrix} \otimes \begin{bmatrix}1\\0\end{bmatrix} = \begin{bmatrix}0\\0\\1\\0\end{bmatrix}; \quad |11\rangle = \begin{bmatrix}0\\1\end{bmatrix} \otimes \begin{bmatrix}0\\1\end{bmatrix} = \begin{bmatrix}0\\0\\0\\1\end{bmatrix}. \end{aligned} \tag{3.32}$$

Thus we have the natural basis for the 4-dimensional vector space from those of two 2-dimensional spaces.

Example 3.5.1. Direct products: to express σ_x on a 2-qubit state as a matrix, we take the direct product of two σ_xs acting on each single qubit state:

$$\sigma_x \otimes \sigma_x = \begin{bmatrix}0 & 1\\1 & 0\end{bmatrix} \otimes \begin{bmatrix}0 & 1\\1 & 0\end{bmatrix} = \begin{bmatrix}0&0&0&1\\0&0&1&0\\0&1&0&0\\1&0&0&0\end{bmatrix}.$$

We can generalize to n qubits: the natural basis of the n-qubit Hilbert space $\mathcal{H}^{\otimes n}$ consists of 2^n orthogonal vectors

$$\{|0\rangle, |1\rangle, |2\rangle, ..., |2^n - 1\rangle\}. \tag{3.33}$$

The interpretation as an n-bit register is straightforward when the labels are written in binary. For example, the 8th basis vector for a 4-qubit Hilbert space will be

$$|7\rangle = |0111\rangle = |0\rangle \otimes |1\rangle \otimes |1\rangle \otimes |1\rangle.$$

The algebra of multi-qubit states generalizes in a natural manner from that of single qubits.

Box 3.8: Algebra of Tensor Product States

Consider two distinct physical systems A and B with Hilbert spaces $\mathcal{H}^A$ of dimensions 2^n and $\mathcal{H}^B$ of dimensions 2^m. Let the basis vectors of these two spaces be denoted $\{|i^A\rangle\},\ i^A = 0, 1, ...2^n - 1$, and $\{|\mu^B\rangle\},\ \mu^B = 0, 1, ...2^m - 1$. If I pick a state $|\phi^A\rangle$ from A and a state $|\psi^B\rangle$ from B, I can form a state in the tensor product Hilbert space $\mathcal{H}^{AB} = \mathcal{H}^A \otimes \mathcal{H}^B$ as

$$|\Phi^{AB}\rangle = |\phi^A\rangle|\psi^B\rangle.$$

- Probability amplitude $\langle i^A, \mu^B|\Phi\rangle = \langle i^A|\phi^A\rangle\langle\mu^B|\psi^B\rangle$
- Inner product $\langle\Phi_1|\Phi_2\rangle = \langle\phi_1^A|\phi_2^A\rangle\langle\psi_1^B|\psi_2^B\rangle$
- Basis states for $\mathcal{H}^{AB}$ is the set of product basis vectors $\{|v_{i\mu}\rangle = |i^A\rangle|\mu^B\rangle\}$
- The most general state in $\mathcal{H}^{AB}$ is a linear combination of these basis states:
$$|\Psi\rangle^{AB} = \sum_{i\mu} C_{i\mu}|v_{i\mu}\rangle.$$
- If two operators $\hat{A}$ and $\hat{B}$ act on each space independently then the action on the product space is given by the operator $\hat{C} = \hat{A} \otimes \hat{B}$.

Summary: The Math and the Physics

The arena of quantum mechanics is the Hilbert space $\mathcal{H}$, the state vectors live here, and transformations of the state vector are operators in $\mathcal{H}$. To be able to work efficiently with the maths, we summarize the correspondence between the mathematical concept and the physical quantities in Table 3.2. Note that this is for “pure” states of isolated quantum systems. (We will discuss mixed states of systems that are influenced by some environment in a later chapter.)

Problems

3.1. Prove that a Hermitian matrix has real eigenvalues and its eigenvectors corresponding to distinct eigenvalues are orthogonal to each other.

3.2. Prove that a unitary matrix has complex eigenvalues of unit magnitude, and that its eigenvectors corresponding to distinct eigenvalues are orthogonal.

TABLE 3.2: Correspondence between the math and the physics of quantum mechanics.

Math	Physics
Normalized vector $\lvert\psi\rangle \in \mathcal{H}$	pure state
Hermitian operator $\hat{A}$ on $\mathcal{H}$	physical observable
Eigenvalues $\{a_i\}$ of $\hat{A}$	set of all possible values obtainable on measuring the observable A
Eigenvector $\lvert a_i\rangle$ of $\hat{A}$	state in which measuring A gives a value a_i
Computational basis $\{\lvert i\rangle\}$, $i = 0,\ 1, 2...$	eigenstates of a suitable fiducial observable
Inner product $\langle i\lvert\psi\rangle$	probability amplitude for the state $\lvert\psi\rangle$ to be in the basis state $\lvert i\rangle$
Amplitude squared $\lvert\langle a_i\lvert\psi\rangle\rvert^2$	probability of obtaining the value a_i on measuring A in the state ψ
Matrix element $A_{ij} = \langle i\lvert A\rvert j\rangle$	amplitude for producing a transition from $\lvert j\rangle$ to $\lvert i\rangle$ by the action of A (No assumption is made here about the nature of the operation)
Diagonal element $\langle\psi\lvert A\rvert\psi\rangle$	average value of the observable A in the state $\lvert\psi\rangle$
Unitary operator $\hat{U}$	possible evolution operator that changes the state reversibly

3.3. Show that if $\hat{H}$ is a Hermitian operator then $e^{i\hat{H}}$ is a unitary operator.

3.4. Given a unitary operator $\hat{U}$, show that the operator $i(\mathbb{1} + \hat{U})(\mathbb{1} - \hat{U})$ is Hermitian.

3.5. For a Hermitian or unitary matrix, show that the sum of diagonal elements (the trace) equals the sum of the eigenvalues, and the determinant equals the product of the eigenvalues.

3.6. For each of the following matrices, find if they are unitary or Hermitian or neither. Find their eigenvalues and eigenvectors. Find if their eigenvectors are orthogonal.

(a) $\begin{bmatrix} 1 & i \\ i & -1 \end{bmatrix}$ (b) $\begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$

3.7. For the three Pauli matrices σ_x, σ_y, and σ_z,

(a) Show that $\sigma_i^2 = \mathbb{1}$.

(b) Show that σ_i's are Hermitian as well as unitary.

(c) Find the commutator $[\sigma_i, \sigma_j] = \sigma_i\sigma_j - \sigma_j\sigma_i$.

(d) Find the *anti*-commutator $\{\sigma_i, \sigma_j\} = \sigma_i\sigma_j + \sigma_j\sigma_i$.

3.8. Show that all the eigenvalues of any projection operator are either 1 or 0.

3.9. Show that the operator which performs a transformation from the Z basis to the X basis has the following matrix representation:

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

This operator is also known as the Hadamard operator and is very useful in quantum computation.

Verify that this operator is Hermitian. Show that it can be expressed as a linear combination of the Pauli matrices.

3.10. Show that for any two operators A and B,

$$AB = \frac{1}{2}[A, B] + \frac{1}{2}\{A, B\}.$$

3.11. Given a unit vector $\hat{\boldsymbol{e}} = (e_x, e_y, e_z)$ in an arbitrary direction, we can define the component of spin along $\hat{\boldsymbol{e}}$ by

$$\sigma_e = e_x\sigma_x + e_y\sigma_y + e_z\sigma_z.$$

(a) Show that $\sigma_e^2 = \mathbb{1}$.

(b) Find the eigenvalues and eigenvectors of σ_e.

3.12. Define a "vector matrix" $\vec{\boldsymbol{\sigma}} = \hat{\boldsymbol{\imath}}\sigma_x + \hat{\boldsymbol{\jmath}}\sigma_y + \hat{\boldsymbol{k}}\sigma_z$. Show that

$$(\vec{\boldsymbol{a}}. \cdot \vec{\boldsymbol{\sigma}})(\vec{\boldsymbol{b}} \cdot \vec{\boldsymbol{\sigma}}) = (\vec{\boldsymbol{a}} \cdot \vec{\boldsymbol{b}})\mathbb{1} + i(\vec{\boldsymbol{a}} \times \vec{\boldsymbol{b}}) \cdot \vec{\boldsymbol{\sigma}} \tag{3.34}$$

for vectors $\vec{\boldsymbol{a}}$ and $\vec{\boldsymbol{b}}$.

3.13. Find the expectation value of σ_e in the state $|0\rangle$. Generalize this result to find the expectation value of σ_e in a state $|\hat{\boldsymbol{f}}+\rangle$ where $\hat{\boldsymbol{f}}$ is a general direction making angle θ with the $\hat{\boldsymbol{z}}$ axis.

Chapter 4

Properties of Qubits

The mathematical foundations built on motivation from experiments being in place, we now look at the properties of qubits that distinguish them from classical bits, and also which make their manipulation and behavior somewhat counter-intuitive.

In this chapter we will list some properties of a quantum system that are peculiar and touch on the so-called weirdness of quantum mechanics. We will also take a peek into the foundational aspects of the theory that are being battled out to this day.

4.1 The Bloch Sphere Representation of a Qubit

A generic qubit could have a non-definite state expressed as a superposition

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \qquad |\alpha|^2 + |\beta|^2 = 1.$$

How do we picture a qubit? As a vector in Hilbert space, the description is abstract. The 2-d Hilbert space is a space with 4 dimensions. To get a better feel for the sort of vector a quantum state is, we look at a geometrical visualization of a qubit.

The space of all possible single qubits is spanned by all values of the four real numbers defined by α and β but subject to the constraint of normalization: $|\alpha|^2 + |\beta|^2 = 1$. We have an additional constraint in the form of equivalence of all states differing by an overall phase. The four parameters thus reduce to two, which determine the surface of a unit sphere in the space of parameters. Let's see how.

Recall the representation of $|\psi\rangle$ in polar form (Equation 3.5):

$$|\psi\rangle = r_1|0\rangle + r_2 e^{i\phi}|1\rangle,$$

where we've written $\phi = \theta_2 - \theta_1$, the relative phase between the basis vectors. We can further parametrize r_1 and r_2 in terms of a single angle θ'

$$r_1^2 + r_2^2 = 1 \implies r_1 = \cos\theta', r_2 = \sin\theta'.$$

We now have

$$|\psi\rangle = \cos\theta'|0\rangle + \sin\theta' e^{i\phi}|1\rangle,$$

which is the standard representation of a point on the unit sphere by spherical polar coordinates $\theta' \in [0, \pi]$ and $\phi \in [0, 2\pi]$.

But we still have one further condition, which is often not intuitively obvious. For a given state at (θ', ϕ), consider the point on this sphere that is diametrically opposite: i.e., at $(\pi - \theta, \pi + \phi)$:

$$|\psi\rangle_{\text{antipode}} = -\cos\theta'|0\rangle - \sin\theta' e^{i\phi}|1\rangle = -|\psi\rangle,$$

which is physically indistinguishable from $|\psi\rangle$. Thus the upper hemisphere of the sphere is sufficient to represent the states of a qubit, i.e., $\theta' \in [0, \pi/2]$. It is useful to regard this space as still a sphere by replacing the parameter θ' by $\theta/2, \theta \in [0, \pi]$. Geometrically this is visualized as "folding" the lower hemisphere on the upper, to obtain the **Bloch sphere**. The usual sphere is a "double cover" of the Bloch sphere. We finally have a representation of the qubit as a unique point on this sphere (Figure 4.1):

$$\begin{aligned} |\psi\rangle \quad = \quad & \cos\frac{\theta}{2}\,|0\rangle + e^{i\phi}\,\sin\frac{\theta}{2}\,|1\rangle; \\ & 0 \le \theta \le \pi, \quad 0 \le \phi \le 2\pi. \end{aligned} \tag{4.1}$$

The vector

$$\vec{\boldsymbol{p}} \equiv (\cos\phi\sin\theta, \sin\phi\sin\theta, \cos\theta) \tag{4.2}$$

is called the Bloch vector, after a notation invented by Felix Bloch in 1943 to depict the polarization states of light. Note that this sphere is not to be regarded as one in 3-d coordinate space.

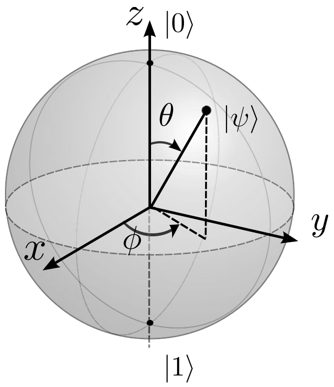


FIGURE 4.1: The Bloch sphere.

On this sphere, the north pole represents $|0\rangle$ and the south pole, $|1\rangle$. In general, antipodal points on the Bloch sphere represent orthogonal state.

This picture is useful for visualizing the effects of single qubit transformations, which would take a point on this sphere to another.

There is no known *simple* generalization of this idea for multiple qubits, but it is useful for testing out ideas on gates and transformations for single qubits.

Exercise 4.1. Using the polar representation for complex numbers α and β, obtain the relationship between the angles θ and ϕ and the magnitude and phase of α and β.

Exercise 4.2. Figure out the location on the Bloch sphere of the states $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ and $\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$.

Exercise 4.3. Show that antipodal states on the Bloch sphere (i.e., those at (θ, ϕ) and at $(\pi - \theta, \pi + \phi)$ are orthogonal.

4.2 Cloning and Deleting

The full specification of a superposition state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ is given by the complex numbers α and β. The meaning of these numbers is physically derived by making measurements on this state, in the computational basis. This process would randomly "collapse" the state to either $|0\rangle$ or $|1\rangle$. The probability of obtaining $|0\rangle$ is $|\alpha|^2$ and of obtaining $|1\rangle$ is $|\beta|^2$. This is true in a statistical sense: make the same measurements on a statistically large set of identically prepared qubits: an ensemble. A measurement on a single qubit state that is unknown projects it on to a basis state and the original state is destroyed.

So if we are given a single quantum system in the state $|\psi\rangle$ then can we make clones (that is, exact copies) of the state so that we can gather the requisite measurement data? The answer given by quantum mechanics is "*NO*".

There exists no quantum mechanical way (i.e., a unitary operator) to take one unknown state and make multiple identical copies of it.

This is the **no cloning theorem** first formulated in 1982 [76, 27], which states that an arbitrary quantum system cannot be cloned by a universal unitary transformation. If $\hat{U}_{cl}$ is a unitary cloning machine, then its action would be defined as taking as input the state $|\psi\rangle$ to be cloned along with a "blank" state, say $|0\rangle$, and produce as output the original state and its clone:

$$\hat{U}_{cl}|\psi\rangle|0\rangle = |\psi\rangle|\psi\rangle. \tag{4.3}$$

Quantum mechanics says this is not true for arbitrary $|\psi\rangle$.

Theorem: A unitary transformation cannot make identical copies of an arbitrary quantum state.

Proof. Suppose there does exist a cloning machine as defined by Equation 4.3. Consider its action on two arbitrary quantum states $|\psi\rangle$ and $|\phi\rangle$:

$$\begin{aligned} \hat{U}_{cl}|\psi\rangle|0\rangle &= |\psi\rangle|\psi\rangle, &\text{(4.4a)}\\ \hat{U}_{cl}|\phi\rangle|0\rangle &= |\phi\rangle|\phi\rangle. &\text{(4.4b)} \end{aligned}$$

Take the inner product of (4.4a) with (4.4b),

$$\begin{aligned} LHS &= \langle\phi|\langle 0|\hat{U}_{cl}^{\dagger}\hat{U}_{cl}|\psi\rangle|0\rangle \\ &= \langle\phi|\psi\rangle, \\ RHS &= \langle\phi|\langle\phi|\psi\rangle|\psi\rangle \\ &= \langle\phi|\psi\rangle^2. \end{aligned}$$

The only way $LHS = RHS$ is if $\langle\phi|\psi\rangle = 0$ (they are orthogonal) or if $\langle\phi|\psi\rangle = 1$ (they are identical). Thus a more rigorous statement of the no-cloning theorem would be that non-orthogonal states cannot be cloned by the same unitary operator. □

Another proof is as follows:

Proof. Since $\hat{U}_{cl}$ is linear, its operation on a linear combination of states will be

$$\hat{U}_{cl}(|\psi\rangle + |\phi\rangle)|0\rangle = |\psi\rangle|\psi\rangle + |\phi\rangle|\phi\rangle.$$

However, a cloner of the state $|\psi\rangle + |\phi\rangle$ must produce

$$(|\psi\rangle + |\phi\rangle)(|\psi\rangle + |\phi\rangle) = |\psi\rangle|\psi\rangle + 2|\psi\rangle|\phi\rangle + |\phi\rangle|\phi\rangle,$$

which is NOT what $\hat{U}_{cl}$ produced! In fact, the output of the cloner is actually an ENTANGLED state (Section 4.4) while what we require is a product state. Due to this inconsistency, $\hat{U}_{cl}$ does not exist. □

You will see an illustration of this using CNOT operations in Chapter 7.

The converse of this theorem is also true. Sometimes referred to as the **no deletion** theorem [52], this states that given multiple copies of an unknown quantum state, no unitary transformation can delete one of the copies to give a blank ($|0\rangle$). This theorem thus protects the information content in a qubit. Both these theorems are of great importance in the theory of quantum information.

4.3 Distinguishability of Qubit States

Classically, the outcomes of decision processes are always distinguishable: it is taken for granted that a tossed coin will land either on heads or on tails and upon looking at it, we can distinguish the different outcomes with certainty. In applications to quantum information processing too, we will usually measure the output state after a process. If this state is to give us answers to the problem we are trying to solve, it is important to be able to distinguish alternate outcomes. In quantum, basis states can get transformed to superpositions. Alternate outcomes may be possible that must be distinguishable. It is easy to see that this is possible if the states are orthogonal.

Suppose the possible final states are $|\psi_1\rangle$ and $|\psi_2\rangle$ that are not orthogonal, $\langle\psi_2|\psi_1\rangle \neq 0$. This means that one can write the second state in terms of the first and its orthogonal complement $|\psi_1\rangle_\perp$:

$$|\psi_2\rangle = a|\psi_1\rangle + b|\psi_1\rangle_\perp.$$

Thus on measuring the output, there is a probability $|a|^2$ that we get $|\psi_1\rangle$ even if the output state being measured was $|\psi_2\rangle$. There is a probability $|a|^2$ of getting the *wrong* outcome when measuring ψ_2. The two output states assumed here cannot therefore be distinguished reliably. This fact can be proved rigorously by showing that one cannot invent any measurement operator that gives distinct outcomes with certainty on measuring a set of states that are not mutually orthogonal. This property is exploited in secure quantum key distribution to make the communication safe.

Other means of distinguishing non-orthogonal states have been invented in which the space of states is extended, and the notion of measurement is generalized. These so-called *unambiguous state discrimination* techniques allow for the possibility of getting inconclusive results after measurement. However if positive results are obtained then they do tell the two states apart.

4.4 Entanglement

We now discuss in detail one of the most startling and yet most useful aspects of superposition.

The most general n-qubit state would be a superposition

$$|\psi\rangle_n = \sum_{x=0}^{2^n-1} \alpha_x \, |x\rangle_n, \quad \sum_x |\alpha_x|^2 = 1, \tag{4.5}$$

where the subscript n is to emphasize that we have an n qubit state. Now it is possible to construct higher-dimensional states by taking direct products of lower-dimensional states. However not all higher-dimensional states can be constructed this way. There will always exist states that cannot be expressed as a direct product. Such states are called **entangled** states. This nomenclature is due to Erwin Schrödinger who first discovered the implication of such states in 1935 [61].

For example, consider two generic qubits

$$|\psi_1\rangle = \alpha_1|0\rangle + \beta_1|1\rangle, \quad |\psi_2\rangle = \alpha_2|0\rangle + \beta_2|1\rangle. \tag{4.6}$$

If you form the direct product, you get

$$|\psi_1\rangle \otimes |\psi_2\rangle = \begin{bmatrix}\alpha_1\\ \beta_1\end{bmatrix} \otimes \begin{bmatrix}\alpha_2\\ \beta_2\end{bmatrix} = \begin{bmatrix}\alpha_1\alpha_2\\ \alpha_1\beta_2\\ \beta_1\alpha_2\\ \beta_1\beta_2\end{bmatrix}. \tag{4.7}$$

This is called a **product state**. Now the most general 2-qubit state is a superposition of the form

$$|\phi\rangle_2 = c_0|0\rangle + c_1|1\rangle + c_2|2\rangle + c_3|3\rangle. \tag{4.8}$$

Equation 4.7 is of a special form:

$$c_0c_3 = c_1c_2. \tag{4.9}$$

Not all states satisfy this property. Those states which do **NOT** are called **entangled states**. Equation 4.9 is the criterion for a 2-qubit state to be a product state.

For example, the state $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ is entangled while $\frac{1}{\sqrt{2}}(|00\rangle + |01\rangle)$ is not. A state like $|00\rangle + |10\rangle + |11\rangle$ is *partially* entangled.

Box 4.1: Bell States

The classic examples of entangled states are the Bell states, so named in honor of John Bell [5] whose famous arguments resolved the Einstein–Podolsky–Rosen paradox [31] involving entangled states. They are also referred to as EPR states for this reason. These states exhibit maximum correlation or anticorrelation between their components:

$$|\beta_{00}\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle); \tag{4.10a}$$

$$|\beta_{01}\rangle = \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle); \tag{4.10b}$$

$$|\beta_{10}\rangle = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle); \tag{4.10c}$$

$$|\beta_{11}\rangle \quad = \quad \frac{1}{\sqrt{2}}\left(|01\rangle - |10\rangle\right) \tag{4.10d}$$

In the states $|\beta_{00}\rangle$ and $|\beta_{01}\rangle$, the spin values of each component are always the same (correlated), while they are always opposite (anticorrelated) for the other two states.

Verify that these states are mutually orthogonal. They can thus be used as a basis for the 2-qubit Hilbert space $\mathcal{H}^2$.

When you have more than two qubits, you can have entanglement between all or some of the component qubits. In a 3-qubit system, for example, you could have entanglement between all three:

$$|\psi_3\rangle = \frac{1}{\sqrt{3}}\left(|010\rangle + |101\rangle\right), \tag{4.11}$$

which is one of the so-called GHZ states (after Greenberger, Horne and Zeilinger [39]). Note for this particular state that each of the component qubits are anticorrelated, with the first and third having the opposite anticorrelation as the second.

You could have entanglement between two qubits alone, for example:

$$|\psi_{12}\rangle = \frac{1}{\sqrt{3}}\left(|000\rangle + |110\rangle\right) \tag{4.12}$$

One can imagine more possible combinations of partial entanglement. Thus for larger dimensional systems, entanglement becomes more complicated.

Entangled states are just some among the possible states of higher dimensional quantum systems. Why do we single them out for a special name and status? What does it mean for a state to be entangled? We have already pointed out that entangled states have properties that make them correlated to each other. When two (or more) systems are in an entangled state, each component system does not have a definite state. This is what it means to say that the superposition cannot be written as a product of states of the component systems.

Let us examine the meaning of correlations in the context of a two-qubit system in the entangled spin state

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

Assume we have a beam of atom pairs in this state, and that we separate each pair carefully without changing the state and send one atom each to Alice and Bob, who proceed to measure the S_z value on their atom. Each has equal probability of having a value ±1/2. Suppose Alice measures a value $+1/2$ on her atom. This means its state has collapsed to $|0\rangle$. But this is possible only if the combined state collapses to $|00\rangle$, so that Bob's atom also collapses to $|0\rangle$. This happens even without Bob making a measurement on his atom. If Bob now measures S_z, he will get a value $+1/2$. Similarly, had Alice obtained $-1/2$, Bob would also measure the same value. There is perfect correlation between the spins of the two particles. Alice and Bob can verify this by making measurements on a large number of qubit pairs in the same state and comparing the values. As another example, if the state were the so-called singlet state

$$|\beta_{11}\rangle = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle,$$

and the same experiment is performed, then there is perfect *anti*correlation between the spins of the two qubits.

In contrast, suppose that the spins were in the state

$$|\psi\rangle_2 = \frac{1}{\sqrt{2}}(|00\rangle + |10\rangle).$$

It's easy to see that this state can be expressed as

$$|\psi\rangle_2 = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)|0\rangle,$$

decomposed into a product of states of each spin. In this un-entangled state, each spin does possess a definite state. The superposition in the state of the first spin is merely a basis state in another basis: the S_x basis. Here there is no correlation between spin measurements made by Alice and those obtained by Bob.

4.4.1 Quantum vs. classical correlations

In what way are the quantum correlations in an entangled quantum state different from correlations in a classical system? If a measurement of a quantum state yields a probabilistic outcome, could we not assume that the observable measured has a definite value that was merely uncovered by the measurement? Then the probabilities encoded in a quantum state would be like classical probabilities, in that they indicate the lack of knowledge we have about the system. The correlations we just saw in the entangled pair would be just like those in classical systems. For instance, say I have a bag with pairs of socks of random colors, each in paired, unlabelled packets. Now suppose you pull out a packet at random, and give one of the pair each to Alice and Bob. If Alice finds she got a red sock then immediately she can tell that Bob has a red sock too! Perfect correlation! As another example, if Alice found a left sock then she knows Bob has a right sock, without Bob looking at his sock: perfect anticorrelation.

In what way is this (anti)correlation different when we talk about a pair of quantum particles in a Bell state? Can it not be that the particles simply possess spin values that are the same (or opposite, depending on the state), and the measurements just discover these values? The question here is a subtle one: we will ask it again in a different way. Can these correlations between entangled particles be explained by some hidden properties that are not evident in quantum theory, that are assigned specific values at the time of production?

This feature has been thoroughly examined by many scholars. Most practising physicists follow the practical school of thought, known as the **Copenhagen School**, so-called after the city of famous physicist Niels Bohr, its main proponent. According to this school, Nature is nothing more than the experimental results, and there is no place for assumed hidden properties. In other words, *the spin of the atom in the entangled pair doesn't have an objective existence until brought into being by a measurement.*

4.4.2 The EPR paradox

A famous 1936 paper by Einstein, Podolsky, and Rosen [31] brought the whole matter to a head. Popularly known as EPR, they examined a thought experiment with entangled particles[1] and concluded that the quantum mechanical description of nature is incomplete, or else a paradox arises. Niels Bohr countered their claim in a paper of the same title with his pragmatic view that there is no more to nature than what quantum mechanics says about it.

It is insightful to examine a simplified version of the EPR thought experiment (due to Bohm [13], and sometimes referred to as "EPRB") to see what they meant. This version of the experiment actually can be, and has been performed subsequently in the laboratory, so we have a concrete handle on the issue.

Consider a source that produces pairs of qubits in the anticorrelated Bell state

$$|\beta_{11}\rangle = \frac{1}{\sqrt{2}}(|0_1 1_2\rangle - |1_1 0_2\rangle). \tag{4.13}$$

Suppose these particles fly off in two different directions and Alice captures one of them while Bob gets the other. Our two experimenters can measure the spin of their particle using SG machines oriented along any desired axis. The source emits many entangled pairs, and the measurements are repeated a large number of times, recorded, and then compared. Let's label the direction along which Alice's detector (SG_a) is oriented as $\hat{\boldsymbol{a}}$ and that of Bob (SG_b) as $\hat{\boldsymbol{b}}$ (Figure 4.2).

Suppose Alice and Bob decided to align their detectors along the *same* direction, $\hat{\boldsymbol{a}} = \hat{\boldsymbol{b}} = \hat{\boldsymbol{z}}$, and recorded their measurements. A selection is made of those pairs for which Alice measured $+1$. The entanglement in the state (Equation 4.13) implies that for each of those pairs, Bob must have measured

[1] It is after this work that Bell states come to be known as EPR pairs.

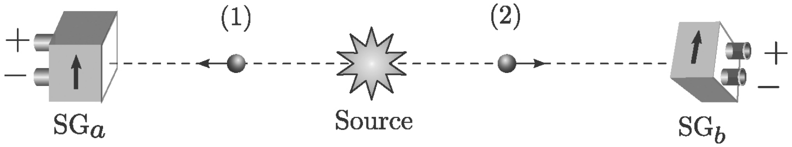


FIGURE 4.2: Spin measurement on an entangled pair of particles.

−1. But this is true even if $\hat{\boldsymbol{a}} = \hat{\boldsymbol{b}} = \hat{\boldsymbol{x}}$ or $\hat{\boldsymbol{y}}$. If we try to explain the correlations by saying that the values were pre-existing before measurement, then we run into problems with the indeterminacy principle! The spins of the individual particles in the pair have fixed (anticorrelated) values in all directions in this scenario, contradicting the quantum mechanical fact that spins in three mutually perpendicular directions are incompatible observables.

EPR has another objection to the quantum dictum that the particles do not have definite values of spin until a measurement is made. Assume at the beginning, a state in which neither qubit has a definite spin. When Alice makes an SG$_a$ measurement, then the combined state collapses to one that is an eigenstate of $\sigma_a \otimes \mathbb{1}$. The collapsed state is also an eigenstate of $\mathbb{1} \otimes \sigma_a$ with the opposite eigenvalue. This conveying of information about the collapse from one qubit to the other is mystifying, particularly if we recall that the two detections are taking place at spatially separated points, and could be really really far from each other! Is this even compatible with Einstein's special relativity, which claims that information cannot travel faster than the speed of light? In such a scenario, it wouldn't even make sense to decide *which* measurement was made first!

Einstein, Podolsky and Rosen summarized their conclusions as follows: The assumption that the quantum system possesses certain properties, *viz.* spin, independent of whether it is measured or not, is known as **realism**. Also, the value of this property cannot be altered by measurements made at spatially separate locations. This tenet is known as **locality**. The EPR experiment shows that quantum mechanics violates **local realism**. In order to sort out the paradox, they concluded that

I. There is some instantaneous mechanism by which the measurement result of the particle at A is conveyed to the experiment at B, meaning that quantum mechanics is *non-local*

OR

II. The quantum mechanical description of the initial entangled state by the vector $|\beta_{11}\rangle$ is *incomplete*, since it does not provide a full specification of the actual system, i.e., it is not *realistic.*

Since non-locality was counter to relativity, EPR were inclined to choose the latter option in claiming that quantum mechanics was incomplete, and some better theory, which had in it variables of the system that were hidden to our view so far, must be correct.

As for the construction of local, hidden variable theories, the death knell was sounded when John Bell [5] showed that any such theory must obey certain inequalities that are NOT obeyed by certain quantum entangled states! We'll take a brief look into these in what follows.

4.4.3 Bell's inequalities and non-locality

Bell's original work, and many subsequent variants show how quantum correlations in an entangled state are essentially different from classical ones. One of the inequalities of Bell applies to a physical system consisting of two subsystems, obeying the principle of local realism. He shows that the quantum statistics for such a system involving entangled subsystems will necessarily violate this inequality, a statement generically known as "Bell's theorem" [64]. Subsequently many similar inequalities were discovered by various authors. (These are reviewed in [18].) We will discuss one of them (not Bell's original one!) to show how quantum correlations are intrinsically different from classical (local realist) ones. This follows original work by Clauser, Horne, Shimony and Holt [17](CHSH).

We consider spin as an example but the derivation holds true for any dichotomic variable, i.e., one with measurements outcomes described by two values, ±1. Let's revisit the experiment of Figure 4.2.

Consider a source emitting a very large number N of entangled spin-half pairs, and four arbitrary directions $\hat{\boldsymbol{a}}, \hat{\boldsymbol{a'}}, \hat{\boldsymbol{b}}, \hat{\boldsymbol{b'}}$ for SG machines chosen by Alice and Bob for measuring. Suppose that before measurement, the spin of the i^{th} pair has hidden, fixed values $r_i(a)$ and $r_i(a')$ for particle (1), $s_i(b)$ and $s_i(b')$ for particle (2) along the respective axes. The correlation between particles (1) and (2) can be measured by the average value of the product of spin measurements:

$$C(a,b) = \frac{1}{N}\sum_i r_i(a)s_i(b). \tag{4.14}$$

We will have similar expressions for $C(a',b), C(a,b')$, and $C(a',b')$, if the experiments used those pairs of axes for measurement. These expressions for the average are the same as for classical statistical averages.

CHSH in their worked aimed to calculate the quantity

$$C(a,b) + C(a,b') + C(a',b) - C(a',b'). \tag{4.15}$$

We'll first see what the "classical" value is, assuming hidden variable description and then compare it with the predictions of quantum mechanics. First look at the possible combinations of spin values (in units of $\hbar/2$) for the i^{th} pair. We introduce the notation

$$T_1 = r_i(a)[s_i(b) + s_i(b')], \quad T_2 = r_i(a')[s_i(b) - s_i(b')].$$

Observe that $T_1 + T_2 = \pm 2$ always. For instance, when $r_i(a) = +1, r_i(a') = -1, s_i(b) = -1, s_i(b') = +1$, then $T_1 = -2$ and $T_2 = 0$. You can see similar results for all other combinations of values for these two spin measurements. To evaluate the sum 4.15, we just sum $T_1 + T_2$ over all i and divide by N:

$$|C(a,b) + C(a,b') + C(a',b) - C(a',b')| \leq 2.. \tag{4.16}$$

This is the CHSH inequality.

What does quantum mechanics predict for the sum (4.15)? Remember that the spins are not to have fixed values before measurement. The correlation between spins are now the quantum mechanical expectation values of spin operator products in the state of Equation 4.13:

$$C(a,b) = \langle \hat{S}_a \hat{S}_b \rangle_{\beta_{11}}. \tag{4.17}$$

Note that the operator $\hat{S}_a$, spin along direction $\hat{\boldsymbol{a}}$ is just $\vec{\boldsymbol{\sigma}} \cdot \hat{\boldsymbol{a}}$ (in units of $\hbar/2$). You would have shown in Problem 3.12 (b) of Chapter 3, that the eigenvectors of $\hat{S}_a$ are given by

$$|\hat{\boldsymbol{a}}\pm\rangle = e^{-i\hat{\boldsymbol{k}}\cdot\vec{\boldsymbol{\sigma}}}|Z\pm\rangle.$$

Here $\hat{\boldsymbol{k}}$ is a direction perpendicular to both $\hat{\boldsymbol{z}}$ and $\hat{\boldsymbol{a}}$, i.e., parallel to $\hat{\boldsymbol{z}} \times \hat{\boldsymbol{a}}$.

Example 4.4.1. Let's find the expectation value of $\hat{S}_a\hat{S}_b$ in the Bell state $|\beta_{11}\rangle$.

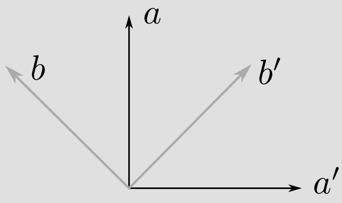
$$\begin{aligned}
\hat{S}_a|\beta_{11}\rangle &= \vec{\boldsymbol{\sigma}} \cdot \hat{\boldsymbol{a}}(|01\rangle - |10\rangle) \\
&= (a_x X_1 + a_y Y_1 + a_z Z_1)(|01\rangle - |10\rangle) \\
&= a_x(|11\rangle - |00\rangle) - i a_y(|01\rangle + |10\rangle) + a_z(|01\rangle + |10\rangle) \\
\hat{S}_a\hat{S}_b|\beta_{11}\rangle &= -a_x b_x|\beta_{11}\rangle - i a_x b_y(|10\rangle + |01\rangle) + a_x b_z(|10\rangle + |11\rangle) \\
&\quad -i a_y b_x(|00\rangle + |11\rangle) - a_y b_y|\beta_{11}\rangle + i a_y b_z(|01\rangle - |10\rangle) \\
&\quad +a_z b_x(|11\rangle + |00\rangle) + i a_z b_y(|01\rangle - |10\rangle) - a_z b_z|\beta_{11}\rangle, \\
\langle\beta_{11}|\hat{S}_a\hat{S}_b|\beta_{11}\rangle &= -a_x b_x - a_y b_y - a_z b_z \\
&= -\hat{\boldsymbol{a}} \cdot \hat{\boldsymbol{b}}.
\end{aligned}$$

Then the left-hand side of Equation 4.16 is

$$\begin{aligned}
|\hat{\boldsymbol{a}} \cdot (\hat{\boldsymbol{b}} + \hat{\boldsymbol{b}'}) + \hat{\boldsymbol{a}'} \cdot (\hat{\boldsymbol{b}} - \hat{\boldsymbol{b}'})| &\leq |\hat{\boldsymbol{a}}||\hat{\boldsymbol{b}} + \hat{\boldsymbol{b}'}| + |\hat{\boldsymbol{a}'}||\hat{\boldsymbol{b}} - \hat{\boldsymbol{b}'}| && (4.18) \\
&= \sqrt{2}(\sqrt{1+\cos\phi} + \sqrt{1-\cos\phi}) && (4.19) \\
\text{where } \cos\phi &= \hat{\boldsymbol{b}} \cdot \hat{\boldsymbol{b}'}. && (4.20)
\end{aligned}$$

Now the minimum value this can take is obviously when $\cos\phi = 0$, and that value is $2\sqrt{2}$, greater than the CHSH bound. Thus there exist configurations of detectors that can violate the CHSH inequality. See for instance Figure 4.3. This leads us to conclude that quantum mechanics is NOT compatible with a local realistic description, that is, the assumption that the spins have values before they are measured must be wrong. The entangled state vector describes the pair as a single whole, with no room for describing the states of the individual constituents. They have no well-defined spin in such a state. There is therefore no way of setting about deriving the CHSH inequality for such a system: the spin values of particles (1) and (2) do not exist before they are measured.

Example 4.4.2. Let's examine the directions for which the CHSH inequality is maximally violated. If $\cos\phi = 0$, then we have $\hat{\boldsymbol{b}} \perp \hat{\boldsymbol{b'}}$. The RHS of inequality 4.18 also shows that $\hat{\boldsymbol{a}}$ and $\hat{\boldsymbol{b}} + \hat{\boldsymbol{b'}}$ must be parallel or antiparallel, and so also $\hat{\boldsymbol{a'}}$ and $\hat{\boldsymbol{b}} - \hat{\boldsymbol{b'}}$ must be parallel or antiparallel. One way of picking such directions is for Alice to choose $\hat{\boldsymbol{z}}$ and $\hat{\boldsymbol{x}}$ while Bob chooses the ±45° directions ($\frac{1}{\sqrt{2}}\left[\hat{\boldsymbol{x}} + \hat{\boldsymbol{z}}\right]$ and $\frac{1}{\sqrt{2}}\left[\hat{\boldsymbol{z}} - \hat{\boldsymbol{x}}\right]$), as in Figure 4.3. Other sets of combinations are also possible that satisfy the above criterion (find them!). In the language of quantum mechanics, we must speak of the operators corresponding to the measurement axes of A and B: in other words, we talk of then measuring the operator σ_a or σ_b. Thus we speak of correlations between certain pairs of observables that violate the CHSH bound for classical correlations.



Example: Alice measures

$\hat{Z}$ and $\hat{X}$

Bob measures

$\frac{1}{\sqrt{2}}(\hat{Z} + \hat{X})$ and $\frac{1}{\sqrt{2}}(\hat{Z} - \hat{X})$

FIGURE 4.3: Directions for SG detectors a, a', b and b' and the corresponding observables measured by Alice and Bob, that maximally violate the CHSH inequality.

The beauty of Bell's inequalities was that for the first time they provided a way to test quantum mechanics experimentally. The first experimental realization of this was performed by the group led by Alain Aspect in 1981 [2]. Since then, many experiments have been performed that confirm the violation of the inequalities, and the corresponding interpretation of quantum mechanics as theory that intrinsically does not obey "local realism".

However, some researchers have tried to come up with non-local theories that still are consistent with relativity, notably the GRW [37] theory of Ghirardi, Rimini, and Weber, and Bohmian mechanics [22]. The debate still continues as people come up with plausible non-local realistic theories to replace quantum mechanics!

This section ought to have convinced you that quantum entanglement is something new and more than classical correlations: leading to its exploitation as a resource in information processing.

Many of the original papers cited in this chapter are reprinted in an invaluable volume by Wheeler and Zurek [72]. A wonderful discussion of many of the properties of quantum systems discussed here is given in the book by Aharonov and Rohrlich [1].

Problems

4.1. Find out what the action of each of the σ_i operators is on the Bloch sphere by checking their effects on the eigenvectors $|Z\pm\rangle, |X\pm\rangle$ and $|Y\pm\rangle$.

4.2. Prove that the Bell states are mutually orthogonal and that they form a basis for $\mathcal{H}^2$. You must be able to express an arbitrary 2-qubit state $|\psi\rangle = a|00\rangle + b|01\rangle + c|10\rangle + d|11\rangle$ as a linear superposition of the Bell states. Find the coefficients in this superposition in terms of a, b, c, and d.

4.3. Entanglement and basis change: suppose $|s_1\rangle$ and $|s_2\rangle$, linear combinations of the basis states $|0\rangle$ and $|1\rangle$ form an orthonormal basis for a spin Hilbert space. Show that the two-spin entangled "singlet" state

$$\frac{1}{\sqrt{2}}(|s_1\rangle \otimes |s_2\rangle - |s_2\rangle \otimes |s_1\rangle)$$

is equivalent to

$$\frac{1}{\sqrt{2}}(|01\rangle - |10\rangle).$$

Check that this preservation of the *form* of entanglement does not hold for the other three Bell states in the transformed basis.

4.4. We found the directions $\hat{\boldsymbol{a}}, \hat{\boldsymbol{a'}}, \hat{\boldsymbol{b}}$, and $\hat{\boldsymbol{b'}}$ of Stern–Gerlach machines for which the CHSH inequality is maximally violated for spin half particles. Translate this experiment to photon polarization measurements and find the corresponding directions for the axes of polarizers used by Alice and Bob that would maximally violate the CHSH inequality.

Chapter 5

Mixed States, Open Systems, and the Density Operator

The formalism for quantum systems developed so far applies to what are called *pure states.* A system in a pure state is completely specified by the state vector. A complete set of experimental tests will determine the system state fully: we have maximal knowledge of the system. For example, for a spin system, we can find a particular orientation of an SG machine such that the state is in its + or − port. This also means that the state is an eigenvector of some operator, or is always a linear combination of the computational basis states.

As opposed to this, as in most practical cases, we only have incomplete knowledge of the state. This means that the state is in practice not an eigenstate of an observable, but consists of a mixture of eigenstates with classical probabilities of being in each state. Such a state is called a **mixed state** and CANNOT be represented by a state vector. The most convenient way of representing and dealing with such systems, is through the density operator formulation, as proposed first by von Neumann [70] in 1927.[1]

For instance, how do we describe the state of an unpolarized beam of spins, such as those emitted from the oven in the original Stern–Gerlach experiment? We will find that on analyzing such a beam using an SG machine in any orientation, it is split into up-spin and down-spin beams of equal intensities. The state of this beam can be regarded as a 50-50 mixture of basis states of any representation. This is an example of a mixed state. We cannot represent it as a superposition of any basis states. However, the output of an SG$_z$ ↑ filter, which splits into up-spin and down-spin beams of equal intensities when passed through SG$_x$ or SG_y machines is a pure state that can be represented by the state vectors

$$|0\rangle \equiv |{\uparrow_z}\rangle = \frac{1}{\sqrt{2}}[|{\uparrow_x}\rangle + |{\downarrow_x}\rangle] = \frac{1}{\sqrt{2}}[|{\uparrow_y}\rangle + i|{\downarrow_y}\rangle ...$$

Another point to keep in mind is that we have so far been describing *closed* quantum systems, that are isolated from the environment or not affected by it. More realistic systems are *open* to the environment, the effect of which must be taken into account in some fashion, though one may not have complete

[1] The density operator was also independently proposed by Lev Landau [45] and by Felix Bloch.

information as to how the environment affects the system. One way of dealing with such situations is to regard the system along with the environment as a big super-system that is closed. So when we concentrate only on the system, we have to average out the effect of the environment. The resulting system state typically is mixed, and one needs the density operator formulation to describe it. The material in this chapter is of a slightly advanced character and may be skipped at first reading.

5.1 The Density Operator

By "state" of a system, we mean a collection of all possible knowledge we can gather about the system, which is practically achieved by studying the distribution of outcomes of measurements made on the system. In the case of pure states, these outcomes together are described by a ray in Hilbert space.

Consider measuring an observable $\hat{Q}$ with N possible eigenvalues q_i with corresponding eigenstates $|q_i\rangle$. If we obtain a particular result q_i, then we can say that the projection operator $\hat{\mathbb{P}}_i = |q_i\rangle\langle q_i|$ has acted on the state of the system.

If we know the state to be the pure state $|\psi\rangle$, then the state is as well described by a projector $|\psi\rangle\langle\psi|$ along this direction. The probability of outcome q_i is given by

$$\begin{aligned} \mathcal{P}(q_i) &= \langle\psi|\hat{\mathbb{P}}_i|\psi\rangle = \langle\psi|q_i\rangle\langle q_i|\psi\rangle \\ &= \langle q_i|\psi\rangle\langle\psi|q_i\rangle \\ &= \langle q_i|\hat{\rho}|q_i\rangle \end{aligned} \tag{5.1}$$

This defines the density operator $\hat{\rho}$ for a pure state described by a single state vector:

$$\hat{\rho}_{\text{pure}} = |\psi\rangle\langle\psi|. \tag{5.2}$$

In general, the system could be composed of a number of (pure) states $|\psi_n\rangle$ where $n = 1, 2...d$, with *classical* probability $p_n : 0 \leq p_n \leq 1, \sum_n p_n = 1$. This mixture $\{p_n, |\psi_n\rangle\}$ is referred to as an *ensemble* of pure states with associated probabilities. In this case, the probability of obtaining the outcome q_i on measuring $\hat{Q}$ is

$$\begin{aligned} \mathcal{P}(q_i) &= \sum_n p_n\langle\psi_n|\hat{\mathbb{P}}_i|\psi_n\rangle = \sum_n p_n\langle\psi_n|q_i\rangle\langle q_i|\psi_n\rangle = \sum_n p_n\langle q_i|\psi_n\rangle\langle\psi_n|q_i\rangle \\ &= \langle q_i|\left(\sum_n p_n|\psi_n\rangle\langle\psi_n|\right)|q_i\rangle \end{aligned} \tag{5.3}$$

where in the third equality, we have moved the term $\langle q_i|\psi_n\rangle$ to the beginning of the expression since it is a number. (This is an illustration of manipulating expressions using the Dirac notation.)

The piece within parentheses in the middle is identified as the **density operator** or the **statistical operator** $\hat{\rho}$ for that state. This operator is completely given by the initial state.

Definition 5.1. *The* ***density operator*** *for a system consisting of a mixed ensemble of states* $\{p_n, |\psi_n\rangle\}$ *is*

$$\hat{\rho} = \sum_n p_n |\psi_n\rangle\langle\psi_n|. \tag{5.4}$$

The sum over states in this expression looks like a superposition of states: but this is an *incoherent superposition*, as opposed to *coherent* superposition of basis states that defines a pure state. The incoherence stems from the fact that the relative phases of the states $|\psi_n\rangle$ are not available to us.

This operator uniquely prescribes the probabilities of outcomes on measurements on the system. Exactly as in Equation 5.1, we can then write the probability of obtaining the outcome q_i as

$$\mathcal{P}(q_i) = \langle q_i|\hat{\rho}|q_i\rangle. \tag{5.5}$$

The expectation value of $\hat{Q}$ in a state $\hat{\rho}$ is

$$\begin{aligned}\langle\hat{Q}\rangle_\rho &= \sum_i q_i\mathcal{P}(q_i) = \sum_i q_i\langle q_i|\rho|q_i\rangle = \sum_{i,j} q_i\langle q_i|q_j\rangle\langle q_j|\rho|q_i\rangle \\ &= \sum_j \langle q_j|\rho\left(\sum_i q_i|q_i\rangle\langle q_i|\right)|q_j\rangle.\end{aligned}$$

Here we have introduced the resolution of identity $\mathbb{1} = \sum_j |q_j\rangle\langle q_j|$ in the third line, and then moved the term $\langle q_i|q_j\rangle$ to the end of the expression since it is a number. In the last line, we identify the term in the parentheses as the spectral representation of the operator $\hat{Q}$.

Definition 5.2. *The* ***trace*** *of an operator* $\hat{A}$ *is defined by*

$$\mathrm{Tr}\hat{A} = \sum_j \langle j|\hat{A}|j\rangle,$$

a simple generalization of the trace of a matrix as the sum of its diagonal elements.

The expectation value of the observable we are measuring is thus given by

$$\langle\hat{Q}\rangle_\rho = \mathrm{Tr}(\rho\hat{Q}). \tag{5.6}$$

(The trace here is apparently taken in the $\{|q_i\rangle\}$ basis, but trace is basis-independent, as you will prove.)

This averaging of the physical property Q is twofold: first the quantum average $\langle Q\rangle_n = \mathrm{Tr}(\rho_n Q)$ over each of the (pure) states comprising the mixture, and the usual statistical average over the whole ensemble with each state average weighted by the probability p_n of its occurrence. We can make this explicit by writing

$$\langle\hat{Q}\rangle_\rho = \overline{\langle\hat{Q}\rangle} = \sum_n p_n \mathrm{Tr}(\rho_{\mathrm{n}}\hat{\mathrm{Q}}). \tag{5.7}$$

Exercise 5.1. Show that for vectors $|\phi_i\rangle$ and $|\phi_j\rangle$, $\mathrm{Tr}(|\phi_i\rangle\langle\phi_j|) = \langle\phi_j|\phi_i\rangle$.

Exercise 5.2. Show that the trace of an operator is independent of the basis chosen to evaluate it.

Exercise 5.3. Show that trace as an operation is linear, i.e., $\mathrm{Tr}(A+B) = \mathrm{Tr}A + \mathrm{Tr}B$ and $\mathrm{Tr}(\lambda A) = \lambda\mathrm{Tr}A$.

Exercise 5.4. Show that the trace of products of operators is invariant under cyclic permutations of the operators. i.e., $\mathrm{Tr}(AB) = \mathrm{Tr}(BA)$, $\mathrm{Tr}(ABC) = \mathrm{Tr}(BCA) = \mathrm{Tr}(CAB)$. etc.

Exercise 5.5. Show that the Pauli matrices are traceless.

The matrix representation of the density operator, called the *density matrix* of the system, is useful for computations. In the computational basis $\{|i\rangle\}$, we can represent the density operator (Equation 5.2) as a matrix:

$$|\psi\rangle = \sum_i c_i|i\rangle \tag{5.8}$$

$$\implies \rho_{\text{pure}} = \sum_{i,j} c_i c_j^*|i\rangle\langle j|. \tag{5.9}$$

For a mixed state, in this basis we can represent the density matrix as

$$\rho_{\text{mixed}} = \sum_{i,j} \rho_{ij}|i\rangle\langle j|. \tag{5.10}$$

If a system consists of equal mixtures of all possible computational basis states it is said to be **maximally mixed**. In n dimensions, such a state is represented by a multiple of the identity matrix:

$$\rho_{\max} = \frac{1}{n}\mathbb{1}_{n\times n}.$$

Example 5.1.1. The density matrix for the unpolarized electron beam discussed above is the maximally mixed state

$$\rho_m = \frac{1}{2}|0\rangle\langle 0| + \frac{1}{2}|1\rangle\langle 1| = \frac{1}{2}\begin{bmatrix} 1 & 0 \\ 0 & 1. \end{bmatrix} = \frac{1}{2}\mathbb{1}.$$

In contrast, the density matrix for the pure state $|\uparrow_x\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ is

$$\begin{aligned} \rho_p &= \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)\ \frac{1}{\sqrt{2}}(\langle 0| + \langle 1|) \\ &= \frac{1}{2}\ (|0\rangle\langle 0| + |0\rangle\langle 1| + |1\rangle\langle 0| + |1\rangle\langle 1|) \\ &= \frac{1}{2}\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}. \end{aligned}$$

Note the difference, though when beams in either state are passed through an SG$_z$ machine, we get ↑ and ↓ outputs with equal probability! However, when passed through an SG$_x$ machine, the mixed state gives the same result, while the pure state $|\uparrow_x\rangle$ gives only an ↑ beam with probability 1.

Example 5.1.2. We'll see how the usual results regarding experimental measurements follow using density matrices for pure states. Consider the state $|\uparrow_x\rangle$ of the above example. The probability of obtaining $+1$ on measuring σ_z in this state is

$$\mathcal{P}_+ = \langle 0|\rho|0\rangle = [\ 1\ 0\ ]\begin{bmatrix} 1 \\ 0 \end{bmatrix} = \frac{1}{2}.$$

The matrix $\sigma_z\rho = \frac{1}{2}\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix}$, and the expectation value of σ_z is the sum of its diagonal elements $= 0$.

Example 5.1.3. A mixed state need not necessarily be composed of orthogonal states. For example, one could have a mixture containing 20% of the state $|0\rangle$and 80% of the state $|\uparrow_x\rangle$, whose density matrix would be given by

$$\rho = \frac{1}{5}|0\rangle\langle 0| + \frac{4}{5}|\uparrow_x\rangle\langle\uparrow_x| = \frac{1}{5}\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + \frac{2}{5}\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} = \frac{1}{5}\begin{bmatrix} 3 & 2 \\ 2 & 2 \end{bmatrix}.$$

5.1.1 Properties of the density operator

The density operator on a Hilbert space, defined by Equation 5.4 satisfies the following properties:

1. $\hat{\rho}$ is Hermitian.

   Proof:
$$\begin{aligned}\hat{\rho}^\dagger &= \sum_n p_n^* |\psi_n\rangle^\dagger \langle\psi_n|^\dagger \\ &= \sum_n p_n |\psi_n\rangle\langle\psi_n| = \hat{\rho}. \end{aligned} \tag{5.11}$$

2. $\hat{\rho}$ is non-negative, that is, for any vector $|v\rangle$, $\langle v|\hat{\rho}|v\rangle \geq 0$. (This translates to its eigenvalues being non-negative, or $\det(\rho) \geq 0$.)

   Proof:
$$\begin{aligned}\langle v|\hat{\rho}|v\rangle &= \sum_n \langle v|p_n|\psi_n\rangle\langle\psi_n|v\rangle \\ &= \sum_n p_n |\langle v|\psi_n\rangle|^2 \geq 0 \end{aligned} \tag{5.12}$$

   since the right side is a sum of numbers that are always positive or zero.

3. It satisfies $\mathrm{Tr}\hat{\rho} = 1$.
   Proof: In an orthonormal basis $\{|i\rangle\}$,

$$\begin{aligned}\mathrm{Tr}\rho &= \sum_i \langle i| \left(\sum_n p_n |\psi_n\rangle\langle\psi_n| \right) |i\rangle \\ &= \sum_n p_n \sum_i \langle i|\psi_n\rangle\langle\psi_n|i\rangle \\ &= \sum_n p_n \langle\psi_n| \left(\sum_i |i\rangle\langle i| \right) |\psi_n\rangle \\ &= \sum_n p_n \langle\psi_n|\psi_n\rangle = 1 \end{aligned} \tag{5.13}$$

In general, any operator on a Hilbert space satisfying these properties is defined as a density operator and can be used to predict the probabilities of outcomes of measurement on the system, bypassing the state-vector formalism altogether.

Example 5.1.4. For a system described by continuous variables, for example position x, the density operator will be expressed as

$$\rho = \int dx\ dx' \omega(x, x')|x\rangle\langle x'|. \tag{5.14}$$

For a pure state, we will have

$$\rho = \int dx\ dx'\ \psi(x)\psi^*(x')|x\rangle\langle x'|, \tag{5.15}$$

where $\psi(x) = \langle x|\psi\rangle$.

Another property of the density operator that will be useful to us is *convexity*:

Definition 5.3. ***Convexity:*** *A set of operators $\{\hat{\rho}_i\}$ form a convex set if*

$$\rho = \lambda\rho_1 + (1-\lambda)\rho_2, \quad 0 < \lambda < 1, \tag{5.16}$$

for every pair $\rho_1, \rho_2 \in \{\rho_i\}$.

Convexity has a very simple meaning: any two members of a convex set can be connected by a straight line without leaving the set. (See Figure 5.1.)

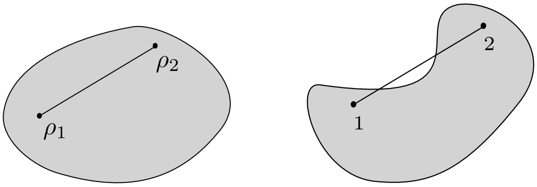


FIGURE 5.1: (a) A convex set, (b) A non-convex set.

5.1.2 Distinguishing pure and mixed states

A given density operator could represent a pure or a mixed state. If the system is pure, then the state is a ray in Hilbert space, and the density operator can be expressed as

$$\rho = |\psi\rangle\langle\psi|, \quad \text{for some } |\psi\rangle.$$

Such a density matrix satisfies

$$\begin{aligned} \rho^2 &= |\psi\rangle\langle\psi|\psi\rangle\langle\psi| = \rho, & (5.17)\\ \mathrm{Tr}(\rho^2) &= \mathrm{Tr}(\rho) = 1. & (5.18) \end{aligned}$$

This is not true if ρ represents a mixed state, where

$$\begin{aligned} \rho &= \sum_n p_n|\psi_n\rangle\langle\psi_n|, \\ \text{for which } \rho^2 &= \sum_{n,m} p_n p_m|\psi_n\rangle\langle\psi_n|\psi_m\rangle\langle\psi_m| \neq \rho. & (5.19) \end{aligned}$$

In the orthonormal basis $\{|i\rangle\}$, we have

$$\begin{aligned}\mathrm{Tr}(\rho^2) &= \sum_i \sum_{n,m} p_n p_m \langle i|\psi_n\rangle\langle\psi_n|\psi_m\rangle\langle\psi_m|i\rangle \\ &= \sum_{i,n,m} p_n p_m \langle\psi_m|i\rangle\langle i|\psi_n\rangle\langle\psi_n|\psi_m\rangle \\ &= \sum n, m p_n p_m |\langle\psi_n|\psi_m\rangle|^2 \\ &\leq \left(\sum_n p_n\right)^2 = 1 \end{aligned} \tag{5.20}$$

The equality holds only when $\langle\psi_n|\psi_m\rangle =$ a pure phase for all pairs n and m, which means that the density matrix comprises only one state vector in Hilbert space: a pure state. In fact, the quantity $\mathrm{Tr}(\rho^2)$ is sometimes called the **purity** of the state. A completely pure state has $\mathrm{Tr}(\rho^2) = 1$ and a completely mixed state has $\mathrm{Tr}(\rho^2) = \frac{1}{n}$. These ideas will be very useful when we study quantum information theory. There we will also encounter the notion of entropy as a measure of information, which can also be used to distinguish pure and mixed states.

Example 5.1.5. For the state pure state $|+\rangle$,

$$\begin{aligned}\rho_p^2 &= \frac{1}{4}\begin{pmatrix}1 & 1\\ 1 & 1\end{pmatrix}^2 = \frac{1}{4}\begin{pmatrix}2 & 2\\ 2 & 2\end{pmatrix} = \rho, \\ \mathrm{Tr}\rho_p^2 &= \frac{1}{2} + \frac{1}{2} = 1.\end{aligned}$$

For the unpolarized electron beam (Example 5.1.1), which is a maximally mixed state, we have

$$\begin{aligned}\rho_m^2 &= \frac{1}{4}\mathbb{1}, \\ \mathrm{Tr}\rho_m^2 &= \frac{1}{2}.\end{aligned}$$

5.1.3 The Bloch ball and the density operator

The representation of a single qubit state on the Bloch sphere can be extended to the density operator. The Bloch sphere is parametrized by spherical angles or in terms of the Bloch vector of Equation 4.2, which characterizes the *polarization* of the state. Can we use this kind of description for a mixed state?

For a single qubit state, ρ is a 2×2 matrix and we can represent it as a linear combination of the Pauli spin matrices and the identity:

$$\rho = a_0 \mathbb{1} + \vec{\boldsymbol{a}} \cdot \vec{\boldsymbol{\sigma}}, \quad \vec{\boldsymbol{a}} \equiv (a_1, a_2, a_3).$$

Since ρ is Hermitian, we need a_0 and a_i to be real. Since $\mathrm{Tr}(\rho) = 1$ and the Pauli matrices are traceless, we must have $p_0 = \frac{1}{2}$. Thus, if $p_i = \frac{1}{2} a_i$, we can write

$$\rho = \frac{1}{2}(\mathbb{1} + \vec{\boldsymbol{p}} \cdot \vec{\boldsymbol{\sigma}}) = \frac{1}{2} \begin{bmatrix} 1 + p_3 & p_1 - i p_2 \\ p_1 + i p_2 & 1 - p_3 \end{bmatrix}. \tag{5.21}$$

Since ρ must be positive, we need $\det \rho \geq 0$.

$$\det \rho = \frac{1}{2}(1 - \vec{\boldsymbol{p}}^2).$$

So ρ is non-negative only if

$$\vec{\boldsymbol{p}}^2 \leq 1, \tag{5.22}$$

with the equality holding for

$$\text{pure states: } |\vec{\boldsymbol{p}}| = 1; \det \rho = 0. \tag{5.23}$$

The vector $\vec{\boldsymbol{p}}$, also referred to as the polarization vector, is a point on or inside the unit sphere: the **Bloch ball**. Thus, states of single qubits can be represented on the Bloch sphere if they are pure and inside the Bloch sphere if they are mixed.

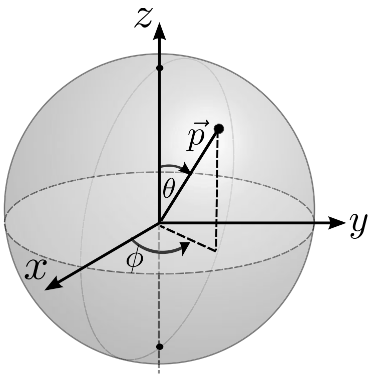


FIGURE 5.2: Bloch ball: points inside the Bloch sphere represent qubits in mixed states

Example 5.1.6. For a pure state, the Bloch representation of the density matrix is of the form

$$\rho = \frac{1}{2}(\mathbb{1} + \hat{\boldsymbol{p}} \cdot \vec{\boldsymbol{\sigma}}),$$

where $\hat{\boldsymbol{p}}$ is the unit polarization vector of the state. To see this, use the Bloch sphere representation of the state vector along the direction $\hat{\boldsymbol{p}} = \{\theta, \phi\}$:

$$\begin{aligned}
\rho(\hat{\boldsymbol{p}}) &= |\psi(\hat{\boldsymbol{p}})\rangle\langle\psi(\hat{\boldsymbol{p}})| \\
&= \begin{bmatrix} \cos\frac{\theta}{2} \\ e^{i\phi}\sin\frac{\theta}{2} \end{bmatrix} \begin{bmatrix} \cos\frac{\theta}{2} & e^{-i\phi}\sin\frac{\theta}{2} \end{bmatrix} \\
&= \begin{bmatrix} \cos^2\frac{\theta}{2} & e^{-i\phi}\cos\frac{\theta}{2}\sin\frac{\theta}{2} \\ e^{i\phi}\cos\frac{\theta}{2}\sin\frac{\theta}{2} & \sin^2\frac{\theta}{2} \end{bmatrix} \\
&= \frac{1}{2}\mathbb{1} + \frac{1}{2}\begin{bmatrix} \cos\theta & e^{-i\phi}\sin\theta \\ e^{i\phi}\sin\theta & -\cos\theta \end{bmatrix} \\
&= \frac{1}{2}(\mathbb{1} + \hat{\boldsymbol{p}} \cdot \vec{\boldsymbol{\sigma}}).
\end{aligned}$$

Exercise 5.6. Calculate the expectation value $\langle \hat{\boldsymbol{n}} \cdot \vec{\boldsymbol{\sigma}} \rangle$ of the spin along the direction $\hat{\boldsymbol{n}}$, in the mixed state characterized by a polarization vector $\vec{\boldsymbol{p}}$ to validate the interpretation of $\vec{\boldsymbol{p}}$ as the polarization along the direction $\hat{\boldsymbol{n}}$.

Exercise 5.7. Locate in the Bloch ball the states given by the following density matrices: (a) $\frac{1}{2}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ (b) $\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$.

5.1.4 Decomposition of the density operator

Often the density operator is the primary descriptor of a state. The decomposition in terms of component states

$$\rho = \sum_i p_i |i\rangle\langle i|,$$

is not always unique.

For a pure state, it must be obvious that there is only one such decomposition, and this can be proved from the definitions:

Theorem 5.1. *For a pure state, there is a* unique *decomposition of $\hat{\rho}$ in the form of Equation 5.4, and in fact that decomposition consists of only one term.*

Proof. We can see this by invoking the convexity property (Equation 5.16). Suppose our pure state density matrix admits such a decomposition

$$\begin{aligned}\hat{\rho}_{\text{pure}} &= \lambda|\psi_1\rangle\langle\psi_1| + (1-\lambda)|\psi_2\rangle\langle\psi_2| \\ &= \lambda\rho_1 + (1-\lambda)\rho_2.\end{aligned}$$

Now since the state is pure, there exists some vector $|u\rangle$ such that

$$\hat{\rho}_{\text{pure}} = |u\rangle\langle u|.$$

Consider an orthogonal vector $|v\rangle : \langle u|v\rangle = 0$.

$$\implies \langle v|\hat{\rho}_{\text{pure}}|v\rangle = \langle v|u\rangle\langle u|v\rangle = 0.$$

$$\implies \lambda\langle v|\hat{\rho_1}|v\rangle + (1-\lambda)\langle v|\hat{\rho_2}|v\rangle = 0.$$

Since λ and $(1-\lambda)$ are positive, this equation can only be satisfied if $\langle v|\hat{\rho_1}|v\rangle = 0 = \langle v|\hat{\rho_2}|v\rangle$. This means $\hat{\rho_1}$ and $\hat{\rho_2}$ are orthogonal to $|v\rangle$. But $|v\rangle$ can be *any* vector orthogonal to $|u\rangle$. So we must have

$$\rho_1 = \rho_2 = \hat{\rho}.$$

□

On the other hand, a mixed state ρ has *no* unique decomposition in terms of pure states! This is easiest to see in our example of an unpolarized beam with density matrix $\frac{1}{2}\mathbb{1}$: on subjecting this beam to SG tests along z or x or y or any other direction $\hat{\boldsymbol{n}}$, it yields equal proportions of $|\uparrow\rangle$ and $|\downarrow\rangle$ states. This means that it can equally well be represented as equal parts of $|0\rangle$ and $|1\rangle$, or $|\uparrow_x\rangle$ and $|\downarrow_x\rangle$ or even $|\uparrow_n\rangle$ and $|\downarrow_n\rangle$!

Exercise 5.8. Show that the density matrix $\frac{1}{2}\mathbb{1}$ can be expressed as $\frac{1}{2}(|\uparrow\rangle\langle\uparrow| + |\downarrow\rangle\langle\downarrow|)$ in any basis.

In fact we can see from the convexity property of density matrices (5.16) that a given density operator ρ can be expressed in infinitely many ways in that form, so that it is impossible to identify any unique component density operators ρ_1 and ρ_2. For example, in the case of a single qubit state, three different decompositions in terms of pure states that sit on the surface of the Bloch sphere, are shown in Figure 5.3. An infinite number of such decompositions is possible by choosing different chords.

Suppose that we prepare a mixed state ρ with pure states $|\psi_n\rangle$ in certain proportions p_n, of the form in Equation 5.4. The p_n's in the density matrix represent the probability of finding the state in $|\psi_n\rangle$. However, when this state is passed on to someone who doesn't know how it was prepared, there is no way they can tell which states were used to prepare the system. Therefore, the p_n's can no longer be interpreted as probability of being in state $|\psi_n\rangle$, since the decomposition is not unique. For this reason, it is not possible to interpret the eigenvalues of a density matrix as physical probabilities of the system being in particular states.

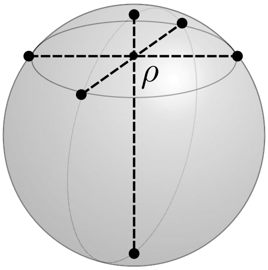


FIGURE 5.3: Density matrix ρ allowing three different decompositions.

5.2 Quantum Mechanics with Density Operators

We now have an alternate formulation of quantum mechanics, in terms of density operators instead of state vectors, that is good for open systems as well. Let's go through the axioms of quantum mechanics framed in this language.

5.2.1 States and observables

Postulate 1. ***Quantum State:*** *The state of a quantum system is described by a density operator in Hilbert space, i.e., a positive Hermitian operator with unit trace.*

Postulate 2. ***Observables:*** *An observable A is represented by a Hermitian operator $\hat{A}$ on Hilbert space. When measured in a state ρ, the probability of an outcome a_n is given by*

$$\mathcal{P}(a_n)_\rho = \mathrm{Tr}(\rho\hat{\mathbb{P}}_n) \tag{5.24}$$

where $\hat{\mathbb{P}}_n = |n\rangle\langle n|$ is the projection on the appropriate eigenspace of $\hat{A}$. The expectation value of the observable is given by

$$\langle\hat{A}\rangle_\rho = \mathrm{Tr}(\rho\hat{A}). \tag{5.25}$$

5.2.2 Generalized measurements

When measurements are made on open systems, we are forced to generalize our notion (from Section 3.3) of projections on the eigenspaces of the observable being measured. Those are special cases and are called *von Neumann* or *projective* measurements.

Most real measurements are not of this kind. To take a simple but extreme example: how do we describe the measurement of the position of a photon in an experiment where it strikes a screen that emits a phosphorescent flash? In this case the position may be noted, but the photon has been absorbed by the screen! Thus we can no longer say the measurement is projective with the post-measurement state being given by Equation 3.17. In fact the photon itself is destroyed by measurement. Another assumption of the projective measurement model is that the measurement is repeatable: successive actions of the projection operator on the same state give the same result. Most real measurements are not repeatable. We therefore need to generalize the idea of measurement.

The main characteristic of any operator representing measurement is that it must tell us how to calculate the probabilities of outcomes. The projective measurements considered in Chapter 3 can be expressed in the density operator formalism as follows. If the outcome is α then the state is transformed by the projection operator $\hat{\mathbb{P}}_\alpha = |\alpha\rangle\langle\alpha|$:

$$\rho \xrightarrow{\text{Measure } \hat{A}, \text{ obtain } \alpha} \hat{\mathbb{P}}_\alpha \rho \hat{\mathbb{P}}_\alpha^\dagger. \tag{5.26}$$

The probability of obtaining the outcome α is given by

$$\mathcal{P}(\alpha) = \mathrm{Tr}(\hat{\mathbb{P}}_\alpha \rho \hat{\mathbb{P}}_\alpha^\dagger) = \mathrm{Tr}(\hat{\mathbb{P}}_\alpha^\dagger \hat{\mathbb{P}}_\alpha \rho) = \mathrm{Tr}(\hat{\mathbb{P}}_\alpha \rho). \tag{5.27}$$

The last step follows from the orthogonality of projection operators (Equation 3.21): $\hat{\mathbb{P}}_\alpha^\dagger \hat{\mathbb{P}}_\alpha = \hat{\mathbb{P}}_\alpha$. It is this property that we drop in the case of generalized measurements.

For generalized measurement, we think in terms of a complete set of measurement operators $\hat{M}_m$, each of which corresponds to a different measurement outcome m. But these operators do not need to be orthogonal like projection operators.

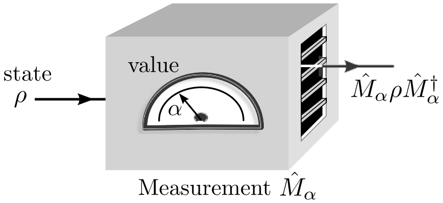

state
ρ
value
α
$\hat{M}_\alpha \rho \hat{M}_\alpha^\dagger$
Measurement $\hat{M}_\alpha$

FIGURE 5.4: Generalized measurement.

Postulate 3. ***Measurement:*** *a measurement process capable of yielding m possible distinct outcomes can be described by a set of Hermitian* measurement operators $\hat{M}_m$ *satisfying* $\sum_m \hat{M}_m^\dagger \hat{M}_m = \mathbb{1}$ *(the completeness relation). The probability of an outcome m is*

$$\mathcal{P}(m) = \mathrm{Tr}(\hat{M}_m^\dagger \hat{M}_m \rho) \tag{5.28}$$

and the state after measurement is given by the density operator

$$\rho_m = \frac{\hat{M}_m \rho \hat{M}_m^\dagger}{\mathrm{Tr}(\hat{M}_m^\dagger \hat{M}_m \rho)}. \tag{5.29}$$

The special case of projective measurements corresponds to $\hat{M}_m^\dagger \hat{M}_m \equiv |m\rangle\langle m| = \hat{\mathbb{P}}_m$.

5.2.3 Measurements of the POVM kind

In most applications of measurement, we are not interested in the post-measurement state of the system, but only in the statistics, or the relative probabilities of different outcomes, that we can collect by measuring an ensemble. A special case of the measurement postulate caters to this need, and is known as the POVM formalism. The set of measurement operators is known as a **positive operator-valued measure**[2] or POVM for short. The reason for this technical-sounding name is not important; we will just describe the main elements of this formalism here, due to its usefulness and pervasiveness in literature.

If we consider the set of operators

$$\hat{E}_m = M_m^\dagger M_m, \qquad \sum_m \hat{E}_m = \mathbb{1}, \tag{5.30}$$

then the probability of outcome m on making a measurement on the state ρ is

$$\mathcal{P}(m) = \mathrm{Tr}(\hat{E}_m \rho).$$

It can be easily seen that the operators $\hat{E}_m$ are positive, but not necessarily orthogonal. That is,

$$\hat{E}_m \hat{E}_n \neq \delta_{mn} \hat{E}_m.$$

They are called the POVM elements, with the set $\{\hat{E}_m\}$ called the POVM. For our purposes, the POVM is just a set of positive operators that add up to unity. Some texts also call these operators as forming a non-orthogonal partition of unity (as opposed to the orthogonal partition made by projection operators).

Example 5.2.1. If we consider the projectors $\mathbb{P}_m = |m\rangle\langle m|$ as the measurement operators, then POVM elements are

$$\hat{E}_m = \mathbb{P}_m^\dagger \mathbb{P}_m = \mathbb{P}_m,$$

the same as the measurements operators themselves. Some texts call these **projection-valued measures** or PVMs.

[2] The word "measure" becomes relevant only in the case of infinite dimensional Hilbert spaces.

Example 5.2.2. One context in which POVM is very useful is in distinguishing two non-orthogonal states with maximum probability. Consider for example the states

$$|\psi_1\rangle = |0\rangle, \quad |\psi_2\rangle = |+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle).$$

The operators $|1\rangle\langle 1|$ and $|-\rangle\langle -|$ project onto orthogonal subspaces. We can form a partition of unity by adding a third operator, so that the set

$$\begin{aligned} \hat{E}_1 &= \left(2-\sqrt{2}\right)|1\rangle\langle 1|, \\ \hat{E}_2 &= \left(2-\sqrt{2}\right)|-\rangle\langle -| \\ \hat{E}_3 &= \mathbb{1} - (\hat{E}_1 + \hat{E}_2). \end{aligned}$$

forms a POVM. Verify that each of the $\hat{E}_i$ is positive.

If we measure these operators, $\hat{E}_1$ and $\hat{E}_2$ giving outcomes yield positive conclusions: there will be no outcome corresponding to $\hat{E}_1$ if the state were $|\psi_1\rangle$, and none corresponding to $\hat{E}_2$ if the state were $|\psi_2\rangle$. But when the outcome corresponding to $\hat{E}_3$ occurs, then we cannot tell which state we had. These operators thus give us a way of unambiguously distinguishing the two states except in the third (inconclusive) case.

The POVM formalism is especially useful when we consider a system in a mixed state as a subspace of a larger system in a pure state. If we perform projective measurements on a larger space, the effect on the subspace is of POVM measurements (see Box 5.1). This is in fact the motivation for a theorem due to Neumark, which states that any POVM can be realized as a projective measurement on an extended Hilbert space.

5.2.4 State evolution

How is the evolution of a system described in terms of density matrices? The evolution operator U for a closed system must be unitary. So for a closed system evolving from initial time $t_0 = 0$ to some final time t, we can write

$$\rho(t) = U(t)\rho(0)U^\dagger(t). \tag{5.31}$$

For a mixed state, $\rho = \sum_n p_n|\psi_n\rangle\langle\psi_n|$. Assuming that time evolution preserves this linearity, we can extend Equation 5.31:

$$\rho(t) = \sum_n p_n U(t)|\psi_n\rangle\langle\psi_n|U^\dagger(t). \tag{5.32}$$

Now the unitary time-evolution operator is obtained from the energy operator, or Hamiltonian $\hat{H}$ for the system:

$$\hat{U}(t) = \exp(-i\hat{H}(t-t_0)/\hbar).$$

Can we write a differential equation for time evolution, like the Schrödinger equation 3.23 for the state vector? If we differentiate Equation 5.31 with respect to time:

$$\begin{aligned}\frac{d}{dt}\rho(t) &= \frac{i}{\hbar}\left(-\hat{H}\hat{\mathcal{U}}\rho(0)\hat{\mathcal{U}}^\dagger + \hat{\mathcal{U}}\rho(0)\hat{H}\hat{\mathcal{U}}^\dagger\right)\\ &= \frac{i}{\hbar}\left(-\hat{H}\rho(t) + \rho(t)\hat{H}\right)\\ &= \frac{i}{\hbar}\left[\rho(t),\hat{H}\right]. \qquad (5.33)\end{aligned}$$

For an open system, the evolution of the density matrix can no longer be expressed as a unitary transformation. The description of such evolution is beyond the scope of this text.

5.3 Composite Systems

There is another sense in which density operators are a useful way to describe nature. In general, it is impossible to isolate the system of interest from some parts of its environment. We then have to regard our system as a subsystem of a larger system: "system + environment". If the large system in a pure quantum state consists of subsystems, then the state of any subsystem is essentially described by a density operator. The way to get there is to perform a *reduction* of the density matrix of the larger system, by a procedure called the *partial trace* over all subsystems except the one of interest.

5.3.1 Reduced density operator

Consider a composite of two systems A and B, described by a pure state density operator ρ^{AB}.

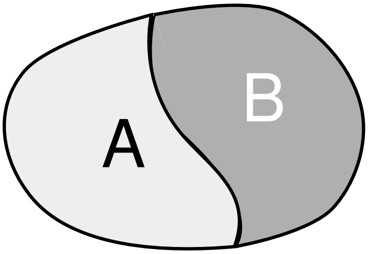


FIGURE 5.5: Illustrating a bipartite composite system.

For the purposes of this book, we will only concentrate on systems consisting of *two* subsystems, the so-called **bipartite** systems (Figure 5.5). We can perform a partial trace over the system B alone to obtain the state of system A. If the set $\{|k^B\rangle\}$ forms a basis for system B then

$$\rho^A = \mathrm{Tr}_B\rho^{AB} = \sum_k \langle k^B|\rho^{AB}|k^B\rangle. \tag{5.34}$$

Trace operation is linear, and if we demand that partial trace is also linear in its inputs, we can compute partial traces in practice.

Definition 5.4. *If subsystems A and B are given by Hilbert spaces spanned by the bases $\{|i^A\rangle\}$ and $\{|j^B\rangle\}$ respectively, we define the partial trace of ρ^{AB} with respect to subsystem A as*

$$\mathrm{Tr}_A\rho^{AB} = \sum_i \langle i^A|\rho^{AB}|i^A\rangle \tag{5.35}$$

which will be an operator on the Hilbert space of subsystem B alone.

Example 5.3.1. Consider a simple example where the system state can be written in separable form:

$$\rho^{AB} = \sigma_1^A \otimes \sigma_2^B.$$

Then quite trivially,

$$\rho^A = \mathrm{Tr}_B(\sigma_1^A \otimes \sigma_2^B) = \sigma_1^A \mathrm{Tr}_B\sigma_2^B = \sigma_1^A.$$

Example 5.3.2. A less trivial case where the two subsystems are entangled, so that the state of the system is a Bell state:

$$\begin{aligned} |\psi^{AB}\rangle &= \frac{1}{\sqrt{2}}\left(|0^A\rangle|0^B\rangle + |1^A\rangle|1^B\rangle\right). \\ \Longrightarrow \rho^{AB} &= |\psi^{AB}\rangle\langle\psi^{AB}| \\ &= \frac{1}{2}\left(|00\rangle\langle 00| + |00\rangle\langle 11| + |11\rangle\langle 00| + |11\rangle\langle 11|\right). \end{aligned}$$

To obtain ρ^A by a partial trace over B, we sandwich each term between the basis states of B and add up:

$$\begin{aligned} \rho^A &= \mathrm{Tr}_B\rho^{AB} \\ &= \langle 0^B|\rho^{AB}|0^B\rangle + \langle 1^B|\rho^{AB}|1^B\rangle \end{aligned}$$

We illustrate the calculation of this by first evaluating the contribution by the first term in ρ^{AB} :

$$\begin{aligned}\langle 0^B| \left(|00\rangle\langle 00|\right) |0^B\rangle &+ \langle 1^B| \left(|00\rangle\langle 00|\right) |1^B\rangle \\ &= |0\rangle\langle 0^B|0\rangle\langle 0|\langle 0|0^B\rangle + |0\rangle\langle 1^B|0\rangle\langle 0|\langle 0|1^B\rangle \\ &= |0\rangle\langle 0| + 0.\end{aligned}$$

Evaluating the other terms similarly, we find that

$$\rho^A = \frac{1}{2}\left(|0\rangle\langle 0| + |1\rangle\langle 1|\right) = \frac{1}{2}\mathbb{1}.$$

Thus the subsystem A is in a maximally mixed state! Similarly,

$$\rho^B = \mathrm{Tr}_A\rho^{AB} = \frac{1}{2}\mathbb{1}.$$

This result is a hallmark of entanglement: though the composite system is in a well-defined state, i.e., its density operator contains maximal information about all measurement outcomes in the state, we can say nothing about measurement outcomes on either of the component subsystems: they are in maximally mixed states.

Exercise 5.9. Calculate the density matrices for both subsystems for the other three Bell states.

Exercise 5.10. Consider a 2-qubit system AB with the density matrix $\rho = \frac{1}{2}|\beta_{00}\rangle\langle\beta_{00}| + \frac{1}{2}|10\rangle\langle 10|$. Compute the reduced density matrices ρ^A and ρ^B.

The fact that the reduced density matrices for entangled systems represent mixed states is generic, and can be used to characterize entanglement. As we have already seen, the reduced density matrices for separable systems will always be pure.

Box 5.1: POVM from Projective Measurements on a Composite System

POVM measurements on quantum systems can be realized as projective measurements on an extended "system+ancilla" Hilbert space. Let's consider a system A that is not interacting with the independent ancilla B. The combined AB system is in a product state that can be represented by the density operator

$$\rho^{AB} = \rho^A \otimes \rho^B.$$

A projective measurement on this state is the action of projection operators

$\hat{\mathbb{P}}_m$ on this state. The probability of outcome m is then

$$\begin{aligned}\mathcal{P}(m) &= \mathrm{Tr}\left[\hat{\mathbb{P}}_m(\rho^A\otimes\rho^B)\right]\\ &= \mathrm{Tr}_A\left[\mathrm{Tr}_B\left(\hat{\mathbb{P}}_m\rho^A\otimes\rho^B\right)\right]\\ &= \mathrm{Tr}_A(\hat{E}_m\rho^A)\end{aligned}\tag{5.36}$$

where the $\hat{E}_m$s are operators on the system A. We can identify the matrix elements of these operators by expressing the above equation in components: using orthonormal bases $\{|i\rangle\}$ for the system A and $\{|\mu\rangle\}$ for the ancilla B,

$$\begin{aligned}\mathrm{Tr}_B\left(\hat{\mathbb{P}}_m\rho^A\otimes\rho^B\right) &= \sum_{ij\mu\nu}(\hat{\mathbb{P}}_m)_{j\nu i\mu}(\rho^A)_{ij}(\rho^B)_{\mu\nu}\\ &= \sum_{ij}(E_m)_{ji}(\rho^A)_{ij}\\ \Longrightarrow\ (E_m)_{ji} &= \sum_{\mu\nu}(\hat{\mathbb{P}}_m)_{j\nu i\mu}(\rho^B)_{\mu\nu}.\end{aligned}$$

It is easy to see that the $\hat{E}_m$s defined this way are complete. Suppose ρ^B is diagonal in the basis $\{|\mu\rangle\}$:

$$\rho^B = \sum_\mu p_\mu|\mu\rangle\langle\mu|,$$

$$\sum_m E_m = \sum_\mu p_\mu\langle\mu|\sum_m\hat{\mathbb{P}}_m|\mu\rangle = \mathbb{1}.$$

5.3.2 Schmidt decomposition

Another useful way of dealing with composite systems, the Schmidt decomposition is about expressing the state of a bipartite system in terms of orthonormal states of the two subsystems.

Theorem 5.2. *If* $\{|u_i^A\rangle\}$ *and* $\{|v_j^B\rangle\}$ *are* orthonormal *sets of vectors in the Hilbert spaces of subsystems A and B, respectively, the state of the combined system can be expressed as*

$$|\psi^{AB}\rangle = \sum_i \lambda_i|u_i^A\rangle|v_i^B\rangle.\tag{5.37}$$

The constants λ_i are called **Schmidt coefficients**, and are non-negative real numbers satisfying $\sum_i\lambda_i^2 = 1$. The number of terms in the expansion is known as the **Schmidt number**. While such an expansion may not in general be unique, the Schmidt number is unique for a given state.

Proof. The Schmidt decomposition theorem (5.37) can be proved by simple results in linear algebra.

Consider a general pure state in the computational basis $\{|i^A\rangle|j^B\rangle\}$:

$$|\psi\rangle \quad = \quad \sum_{ij} C_{ij}|i\rangle|j\rangle.$$

Now the matrix C of complex numbers is a square matrix, and therefore (from results in linear algebra) has a *singular value decomposition* (SVD) of the form $C = UDV$ where D is a diagonal matrix and U and V are unitaries. So we can write

$$|\psi\rangle \quad = \quad \sum_{ij}\sum_{k} U_{ik}D_{kk}V_{kj}|i\rangle|j\rangle.$$

By defining $D_{kk} = \lambda_k, \sum_i U_{ik}|i\rangle = |u_k\rangle, \sum_j V_{kj}|j\rangle = |v_k\rangle$ we get the form of Equation 5.37 for $|\psi\rangle$. □

In terms of density matrices,

$$\rho^{AB} = \sum_i \lambda_i^2 |u_i^A\rangle\langle u_i^A| \otimes |v_i^B\rangle\langle v_i^B|. \tag{5.38}$$

If we perform partial traces on this, we will get

$$\rho^A = \sum_i \lambda_i^2 |u_i\rangle\langle u_i|, \quad \rho^B = \sum_i \lambda_i^2 |v_i\rangle\langle v_i|, \tag{5.39}$$

There are some important take-home points to note here:

- Both the reduced density matrices have the same eigenvalues.
- ρ could have zero eigenvalues and those terms are not present in the expansion above. So the sets $\{|u_i^A\rangle\}$ and $\{|v_j^B\rangle\}$ are not bases for $\mathcal{H}^A$ and $\mathcal{H}^B$, but can be extended to bases by including eigenvectors for the zero eigenvalues.

If the composite system is in a product state, then there is obviously only one term in the Schmidt decomposition. Thus the Schmidt number for product states is always 1. Therefore *an entangled state has Schmidt number* > 1. This is one of the first ways of quantifying entanglement.

Example 5.3.3. Let's find the Schmidt form of some simple states:

- $|\psi_1\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |01\rangle)$. This is a Bell state and is already in Schmidt form. It is clearly entangled: it has 2 terms.

- $|\psi_2\rangle = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle) = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle))$ has only one term: it is a product state.

- $|\psi_3\rangle = \frac{1}{\sqrt{3}}(|00\rangle + |01\rangle + |10\rangle)$. This and more general states are more tricky. We would need to find the SVD of the matrix of coefficients C_{ij} for the state represented in the computational basis.

  In our example, $C = \frac{1}{\sqrt{3}}\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$, which turns out to be directly diagonalizable. So the SVD reduces to diagonalization.

  Eigenvalues are $\lambda_{1,2} = \frac{1}{2\sqrt{3}}(1 \pm \sqrt{5})$ and normalized eigenvectors are:

$$\begin{aligned} \alpha_1 = \begin{bmatrix} \alpha_{11} \\ \alpha_{12} \end{bmatrix} &= \frac{1}{\sqrt{10 + 2\sqrt{5}}}\begin{bmatrix} 1 + \sqrt{5} \\ 1 \end{bmatrix}, \\ \alpha_2 = \begin{bmatrix} \alpha_{21} \\ \alpha_{22} \end{bmatrix} &= \frac{1}{\sqrt{10 + 2\sqrt{5}}}\begin{bmatrix} 1 \\ -(1 + \sqrt{5}) \end{bmatrix}. \end{aligned}$$

  We can see that these eigenvectors are orthogonal, so

$$U = \begin{bmatrix} \alpha_{11} & \alpha_{21} \\ \alpha_{12} & \alpha_{22} \end{bmatrix} = V^\dagger.$$

  The Schmidt vectors are simply

$$\begin{aligned} |u_{1,2}\rangle &= \{\alpha_1, \alpha_2\}, \\ |v_{1,2}\rangle &= \{\alpha_1, \alpha_2\}. \end{aligned}$$

5.3.3 Purification

The notion of Schmidt decomposition immediately leads to a converse construction known as *purification*: given a density matrix ρ^A for a mixed state of a system A, one can construct a supersystem AB of which it is a subsystem, such that $|\psi^{AB}\rangle$ is a pure state, and

$$\rho^A = \mathrm{Tr}_B|\psi^{AB}\rangle\langle\psi^{AB}|. \tag{5.40}$$

The most obvious way to construct a purification of a state (in terms of orthonormal basis $\{|i^A\rangle\}$)

$$\rho^A = \sum_i p_i |i^A\rangle\langle i^A|$$

is to take a copy of this state for system B (in terms similarly indexed orthonormal basis $\{|i^B\rangle\}$, and to construct the pure state

$$|AB\rangle = \sum_i \sqrt{p_i}|i^A\rangle|i^B\rangle. \tag{5.41}$$

It is straightforward to verify that the reduced density matrix $\mathrm{Tr}_B\rho^{AB}$ will give ρ^A:

$$\begin{aligned}\mathrm{Tr}_B\rho^{AB} &= \mathrm{Tr}_B \sum_{ij} \sqrt{p_i}\sqrt{p_j}\left(|i^A\rangle\langle j^A|\right)\left(|i^B\rangle\langle j^B|\right) \\ &= \sum_{ij} \sqrt{p_i}\sqrt{p_j}|i^A\rangle\langle j^A|\langle i^B|j^B\rangle \\ &= \sum_{ij} \sqrt{p_i}\sqrt{p_j}|i^A\rangle\langle j^A|\delta_{ij} \\ &= \sum_i p_i|i^A\rangle\langle i^A| = \rho^A.\end{aligned}$$

Naturally we expect unitary freedom in choosing purifications, and indeed one can show that if there exist two purifications $|AB_1\rangle$ and $|AB_2\rangle$ for the system A, then B_1 and B_2 are related by a unitary transformation:

$$|AB_1\rangle = \mathbb{1} \otimes U_2|AB_2\rangle.$$

Problems

5.1. What are the eigenvalues of the density matrix for the pure state $\alpha|0\rangle+\beta|1\rangle$?

5.2. Calculate the eigenvalues of the 1-qubit density matrix expressed in terms of the Bloch vector.

5.3. Find the reduced density matrices for each subsystem and also the Schmidt decomposition for the state

$$|\psi^{AB}\rangle = \frac{1}{2\sqrt{2}}\left(|0^A\rangle(|0^B\rangle + \sqrt{3}|1^B\rangle) + |1^A\rangle(\sqrt{3}|0^B\rangle + |1^B\rangle)\right).$$

5.4. Consider the 2-qubit density matrix

$$\rho = \frac{1}{8}\mathbb{1} + \frac{1}{2\sqrt{2}}(|01\rangle - |10\rangle).$$

Suppose you measure σ_x for the first qubit and σ_z for the second. What is the probability that they are both $+1$?

5.5. Calculate the eigenvalues of the following density matrices. Which of these represent pure states and which, mixed?

(a) $\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$ (b) $\frac{1}{2}\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$ (c) $\frac{1}{3}\begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix}$ (d) $\frac{1}{3}\begin{pmatrix} 2 & 3 \\ 3 & 1 \end{pmatrix}$.

Chapter 6

Computation Models and Computational Complexity

Now that we have the laws for qubits, we need to develop a system for meaningfully manipulating them. Much of the current paradigm for quantum computing is motivated by classical computation theory, especially the circuit model for computation. In this chapter, we will briefly overview the classical model of Boolean circuit theory, and also some of the theoretical concepts involved in classifying the computational complexity of problems. Roger Penrose [53] has a beautiful account of the history of the theory of computation. Another wonderful book on similar lines is that of Douglas Hofstadter [41], both of which will stimulate you to think along the lines of a computer scientist.

6.1 Computability and Models for Computation

For a long time historically, computation was a matter of actually solving, or finding algorithms to solve, various mathematical problems using mechanical or other algorithms. It was only in the early twentieth century that the process of computation was modelled in mathematical terms, largely in the works of Alan Turing, Alonso Church, Kurt Gödel, and Emil Post. Their efforts were directed at extracting the basic properties of a computational process, independent of the platform on which it was executed.

The first question regarding computation that a theoretician asks is whether or not the given problem is *computable*. What exactly does this mean? If the problem is somehow reduced to the calculation of a function, then is this function computable? In order to meaningfully answer this question without having to examine all possible algorithms designed to compute the function, the famous mathematician Alan Turing came up with a theoretical model computer known as the **Turing Machine**, which is a simplification of your desktop computer to the bare bones.

6.1.1 Turing machine

Turing's abstract computing machine captures the concept of an algorithm to evaluate a function. It can be thought of as a mechanical analogue of an algorithm broken down to its bare bones. Now an algorithm basically takes an input in some symbolic form, performs basic manipulations in steps that may even be recursive and finally finishes up with an output. In the paradigm of a computing machine, the machine has a means of accepting and **reading** an input, a set of **instructions** on what basic steps to perform, which may depend on the output at a previous step. The machine must therefore be able to **move** back and forth over previous steps and **write** out the answer at each step, and **halt** when the process is over. This mechanism of comparing outputs to conditions in the program can be achieved easily by attributing an **internal state** to the machine.

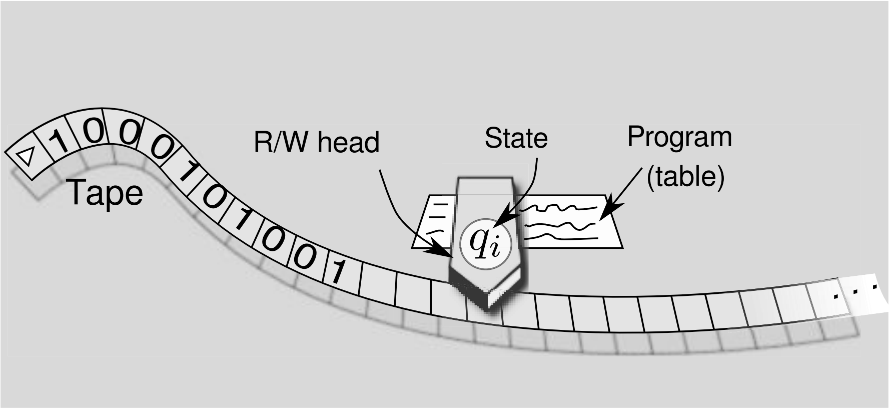


FIGURE 6.1: A schematic of a Turing machine.

Turing modelled this process in an abstract machine (TM), schematically shown in Figure 6.1, consisting of the following.

1. A **tape** which is a string of cells that can contain one of a finite set of symbols, $\Gamma = \{S_i\}$, which could, for example, be binary 0 and 1, a blank ($\emptyset$) and a special symbol $\rhd$, for the left edge of the tape.

2. A **read/write head** that can take input from or write output to a cell at a time when fed into the machine.

3. A **register** that stores the internal state of the machine, which could be one of a finite set of states $\{q_i\}$. There are two special states, S, the starting state and H, the halting state.

4. A **table** of instructions (like a program) that make the head execute a **L**eft move, a **R**ight move, and a **P**rint, depending on the symbol currently read by the head. This is like a function $f(q, x) = \langle q', x', m \rangle$

where q is the current state of the machine, x is the current symbol read, q' is the new state after execution of the step, x' is the symbol written on to the tape, and m is a move L, R, or 0.

Example 6.1.1. To see how a TM might work, consider one with binary symbols, $\Gamma = \{0, 1, \not\emptyset, \triangleright\}$ and internal states $Q = \{S, q_1, q_2, q_3, H\}$. Let the table (program) be as follows:

q_i x	$\triangleright$	0	1	$\not\emptyset$
S	$\langle q_1, \triangleright, R\rangle$			
q_1		$\langle q_1, 0, R\rangle$	$\langle q_1, 1, R\rangle,$	$\langle q_2, \not\emptyset, L\rangle$
q_2		$\langle q_3, \not\emptyset, L\rangle,$	$\langle q_3, \not\emptyset, L\rangle$	
q_3	$\langle H, \triangleright, 0\rangle$	$\langle q_3, 0, L\rangle,$	$\langle q_3, 1, L\rangle$	$\langle H, \not\emptyset, L\rangle$

Can you see what this machine achieves? Take for example an input string 110 on the tape followed by blanks. The tape would look like

$\triangleright$	1	1	0	$\not\emptyset$	...

The sequence of states followed by the machine are:

$$\langle S, \triangleright\rangle \rightarrow \langle q_1, \triangleright\rangle \xrightarrow{R} \langle q_1, 1\rangle \xrightarrow{R} \langle q_1, 1\rangle \xrightarrow{R} \langle q_1, 0\rangle$$
$$\xrightarrow{R} \langle q_2, \not\emptyset\rangle \xrightarrow{R} \langle q_3, \not\emptyset\rangle \xrightarrow{L} \langle q_3, 1\rangle \xrightarrow{L} \langle q_3 \triangleright\rangle \rightarrow \langle H, \triangleright\rangle.$$

The tape now looks like

$\triangleright$	1	1	$\not\emptyset$	$\not\emptyset$	...

You can see that this machine erases the last symbol on the tape. Try it out on a different input.

Exercise 6.1. Try to construct the table of instructions for a TM that adds 1 to the entry on the tape.

Every TM is specified by its own set of symbols Γ, set of internal states Q, and program. So there exists a specific Turing machine for every specific algorithm. However, the machine may be made *programmable* according to different algorithms, if the program is also fed in as part of the input. Thus a programmable Turing machine can simulate any other Turing machine: this is the **universal Turing machine** (UTM).

In his work, strengthened by the work of Alonso Church, who was simultaneously working on Hilbert's famous *computability* problem, Turing was able to prove the thesis that any algorithm could be simulated by a UTM. Church's work strengthened this to the **Church–Turing thesis**:

Theorem 6.1. *Any function that can be computed by an algorithm can be efficiently simulated by the Universal Turing Machine.*

Turing was then able to formulate the problem of computability in terms of whether or not such a universal machine would *halt*, i.e., find a solution. So problems on which this machine halted would then be called computable.

Interestingly enough, the famous **halting problem**, *viz.* whether or not a particular algorithm on a Turing machine will halt, is itself uncomputable!

The question then begged to be asked as to whether a given problem that is not computable by a UTM can be made so by a different paradigm of computation. This led to extensions of the Turing machine concept to probabilistic Turing machines where the algorithms made use of fuzzy logic.

6.1.2 Probabilistic Turing Machine

One of the major challenges to the Church–Turing thesis came from algorithms that were probabilistic, that is, could solve problems efficiently but with a certain (bounded) probability of failure. These problems, for instance the Solovay–Strassen primality test (1977) cannot be efficiently solved on the *deterministic* Turing machine described above.

Computer scientists therefore extended the validity of the Church–Turing thesis to probabilistic algorithms by designing a probabilistic Universal Turing Machine.

A probabilistic or *randomized* Turing machine is one in which randomness is built into each step which chooses possible options according to a probability distribution. Such a machine therefore would need to have an additional tape: the random tape, containing a string of random numbers to decide the options at each step. Without going into details, we will state that a **probabilistic universal Turing machine** (PTM) can replace the earlier deterministic one to save the Church–Turing Thesis in its stronger form.

There is a plethora of randomized complexity classes that can be defined for randomized algorithms that we will not go into, but this extends the class of efficiently solvable problems.

6.1.3 Quantum Turing Machine

The challenge to this model of computation came with trying to simulate quantum mechanical systems on a PTM; this was still unsolvable. The natural question to ask was whether or not it was possible to generalize to a **quantum Turing machine** (QTM) that would further expand the class of solvable problems. This was done by David Deutsch in 1985 though thought of earlier by Benioff and Bennett.

The most important idea behind this machine, which is also probabilistic, is that it is reversible, in the same way as quantum time evolution is reversible.

The idea behind discussing Turing models is to see if the class of problems that are hard to solve can be made smaller. As it turned out, while QTMs cannot reduce the class of unsolvable problems, they do reduce that of hard problems.

The Turing model is the most mathematically abstract model for computation, and is widely used to establish computability and upper bounds on the efficiency of a given algorithm. There are several other models involving for example, decision trees, cellular automata, or logical calculus. The most practical approach is called the circuit model where elementary logic operations are used as building blocks to evaluate the function. It was shown that the circuit model was equivalent to the Turing model, so that there is no loss in generality in concentrating on this, as we will in most of this book.

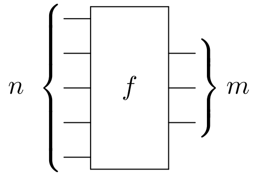
6.2 The Circuit Model and Universal Gates

Classical computation using binary variables works on Boolean logic, and implementation of basic logical operations are done through logic gates that are well known. We will revise their behaviour and notation and express their action as matrix operators.

We will think of a computation as effected by a circuit evaluating some Boolean function whose input is a binary n-bit number, and output may be an m-bit number:

$$f : \{0,1\}^n \mapsto \{0,1\}^m. \tag{6.1}$$

As a circuit this is represented in the following diagram:



The computation is effected by a combination of logic gates. One can represent an n-bit input to a gate as a $2^n \times 1$ column vector and the output as a $2^m \times 1$ column vector. The action of the gate is then represented by a $2^m \times 2^n$ matrix.

A single classical bit takes two mutually exclusive logical values, that can be written as the two basis vectors:

$$0 \equiv \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad 1 \equiv \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \tag{6.2}$$

The logical operation NOT takes a bit and gives its complement: $x \to \bar{x}$. We can algebraically represent this operation of negation as $x \to 1 - x$. Physically it ca nbe implemented by the NOT gate, which flips the value of the input bit, as specified by the truth table and operation

Input	Output
0	1
1	0

$$, \quad \begin{bmatrix}1\\0\end{bmatrix} \to \begin{bmatrix}0\\1\end{bmatrix}, \quad \begin{bmatrix}0\\1\end{bmatrix} \to \begin{bmatrix}1\\0\end{bmatrix}. \tag{6.3}$$

This action can be executed by operation of the following 2 × 2 matrix on either of the bit values:

$$\text{NOT} = \begin{bmatrix}0 & 1\\1 & 0\end{bmatrix}. \tag{6.4}$$

There are more operations possible on two and higher bits. The two-bit numbers are given by 4 column vectors

$$00 \equiv \begin{bmatrix}1\\0\\0\\0\end{bmatrix}, \quad 01 \equiv \begin{bmatrix}0\\1\\0\\0\end{bmatrix}, \quad 10 \equiv \begin{bmatrix}0\\0\\1\\0\end{bmatrix}, \quad 11 \equiv \begin{bmatrix}0\\0\\0\\1\end{bmatrix}. \tag{6.5}$$

A very useful two-bit operation is the AND, which gives an output 1 if an only if both input bits are 1. As a gate, it is given by the truth table and operation

Input	Output
00	0
01	0
10	0
11	1

$$, \quad \begin{bmatrix}1\\0\\0\\0\end{bmatrix} \to \begin{bmatrix}1\\0\end{bmatrix}, \begin{bmatrix}0\\1\\0\\0\end{bmatrix} \to \begin{bmatrix}1\\0\end{bmatrix}, \begin{bmatrix}0\\0\\1\\0\end{bmatrix} \to \begin{bmatrix}1\\0\end{bmatrix}, \begin{bmatrix}0\\0\\0\\1\end{bmatrix} \to \begin{bmatrix}0\\1\end{bmatrix}. \tag{6.6}$$

To represent this operation we need a $2^2 \times 2$ matrix:

$$\text{AND} = \begin{bmatrix}1 & 1 & 1 & 0\\0 & 0 & 0 & 1\end{bmatrix} \tag{6.7}$$

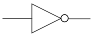
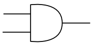
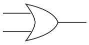
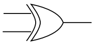
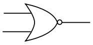
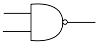
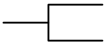
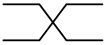
Algebraically, AND can be executed by $(x, y) \to x \wedge y = xy$.

Various other logical operations on two bits are possible, such as the OR, the complements of AND and OR called NAND and NOR respectively, and the exclusive-OR or XOR. These along with their symbols and algebraic equivalents are listed in Table 6.1.

Exercise 6.2. Find the matrix representations of the OR and XOR gates.

In classical computations, we often assume that we work on copies of a

TABLE 6.1: Basic classical gates and their symbols.

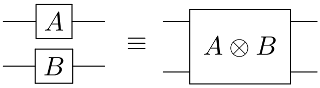
Gate	Logical Symbol	Arithmetic Equivalent	Circuit Symbol
NOT:	$\bar{x}$	$1 - x$	
AND:	$x_1 \wedge x_2$	$x_1 x_2$	
OR:	$x_1 \vee x_2$	$x_1 + x_2 - x_1 x_2$	
XOR:	$x_1 \oplus x_2$	$x_1 + x_2 - 2x_1 x_2$	
NOR:	$x_1 \downarrow x_2$	$1 - x_1 - x_2 + x_1 x_2$	
NAND:	$x_1 \uparrow x_2$	$1 - x_1 x_2$	
COPY:	(fanout)	$x \longrightarrow x, x$	
SWAP:	(crossover)	$x_1, x_2 \longrightarrow x_2, x_1$	

certain input bit, and sometimes inputs are switched. These are included explicitly among the logic gates, as actions we need to perform though we may ignore them at times. This becomes especially important when we map classical functions to quantum ones, because in manipulating qubits, copy can no longer be implemented, and swap is non-trivial.

We can concatenate gates in series to obtain an effective action by multiplying the matrices representing the gates:

$$-\boxed{A}-\boxed{B}- \equiv -\boxed{BA}-.$$

Further, gates could act in parallel in which case the effective action is obtained by taking the tensor product of the corresponding matrices.



In general, the evaluation of a function can be converted to an algorithm involving logic gates acting on the input bits, and a corresponding circuit can be constructed. Now an $n \to m$ function is equivalent to evaluating each of the m outputs as a $\{0,1\}^n \to \{0,1\}$ function. We can therefore restrict our attention to $n \to 1$ functions. Note that for a given n there are 2^{2^n} such distinct functions.

Now we show that any such function can be evaluated using a small subset of the above logic gates: a set of **universal gates**.

6.2.1 Universal gates

It is well known that AND, NOT and OR form a universal set of gates. Consider for example the case $n = 1$. We have four distinct functions implemented as in Table 6.2.

TABLE 6.2: The four 1-bit functions.

Function	Action	Form	Gate
f_1:	$0 \to 0$		
	$1 \to 0$	$f(x) = x \wedge 0$	AND
f_2:	$0 \to 0$		
	$1 \to 1$	$f(x) = x$	Identity
f_3:	$0 \to 1$		
	$1 \to 0$	$f(x) = \bar{x}$	NOT
f_4:	$0 \to 1$		
	$1 \to 1$	$f(x) = x \vee 1$	OR

For $n > 1$, the functions fall into two classes: those giving output 0 and those giving output 1. Suppose for a given function that the output is 1 for the set of inputs $\{x^a\}$. The function can then be constructed in terms of what are called the **minterms** of f, defined as:

$$f^a(x) = \begin{cases} 1, & x \in \{x^a\} \\ 0 & \text{otherwise.} \end{cases} \tag{6.8}$$

The minterms are easily constructed from the bits in the input by the product (AND) of the bits or their complements. For instance, say $x^k = 10110 \in \{x^a\}$. Then

$$f^k(x) = x_5 \wedge \bar{x_4} \wedge x_3 \wedge x_2 \wedge \bar{x_1}. \tag{6.9}$$

We can then construct $f(x)$ as the sum (OR), of the minterms. Then we have the so-called **disjunctive normal form** of $f(x)$:

$$f(x) = f^1(x) \vee f^2(x) \vee \dots \tag{6.10}$$

Thus we need OR, AND, and NOT operations to construct this function. Since we will need more than one copy of the bits in the input to construct the minterms, we require COPY as well.

There is an alternative inductive proof for this. Assume that we have a circuit built only of AND, NOT, and OR gates to construct $f(x)$ for some n. Then to construct an $n + 1 \to 1$ function, we define two $n \to 1$ functions, whose values are given by the output of $f(x)$, as the $(n+1)^{\text{th}}$ bit is set to 0 or 1:

$$f_0(x_n x_{n-1} \dots x_1) = f^{n+1}(0x_n x_{n-1} \dots x_1), \tag{6.11a}$$

$$f_1(x_n x_{n-1} \dots x_1) = f^{n+1}(1x_n x_{n-1} \dots x_1). \tag{6.11b}$$

Then,

$$f(x) = (f_0 \wedge \bar{x}_{n+1}) \oplus (f_1 \wedge x_{n+1}). \tag{6.12}$$

Thus f^{n+1} can be implemented by the circuit of Figure 6.2.

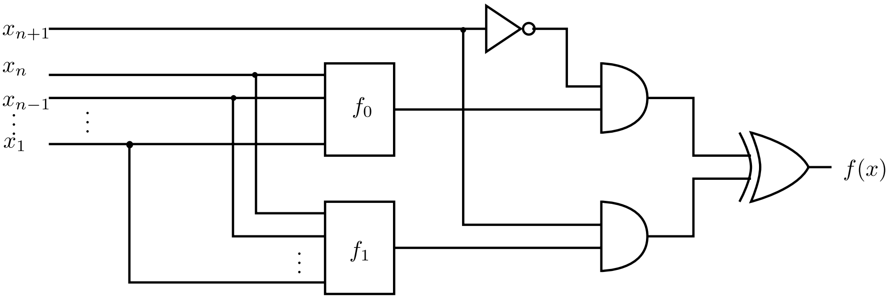


FIGURE 6.2: Classical circuit for function evaluation.

6.3 Reversible Computation

We are studying classical gates to help us develop quantum gates. Quantum gates are unitary. This means they are reversible: they can be "run backward". More practically, the meaning is that the inputs can be deduced from the outputs. Most classical gates however, are irreversible, and cannot as such be extended to quantum gates. For example, the AND gate, being 2 → 1 is irreversible: it gives an output of 0 for more than one input set: $(0,0), (0,1)$, and $(1,0)$. So given only the output, the input cannot be deduced. So is the OR gate and all the other famous 2-bit 2 → 1 gates! For an n-bit gate to be reversible it must at least be a 1 → 1 mapping. Further it must give distinct outputs for different inputs. Thus the outputs are all simply permutations of the inputs. In terms of matrix representations, reversible gates must be invertible. The classical two-bit gates represented by non-square matrices can clearly not be inverted.

The idea of reversibility in classical computation has been studied long before quantum gates were thought of (see for example Bennett [7]). It began with the ideas of Landauer [46], who argued that *erasure* of information is accompanied by a loss in energy. Irreversible gates essentially erase some bits of information in their functioning, and this should lead to intrinsic dissipation of energy. Thus if one wants the most energy-efficient computing machine it should employ reversible gates.

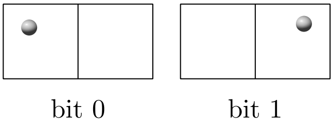


FIGURE 6.3: A simple thermodynamic system encoding a bit of information.

A simple way to understand how erasing information costs is in terms of the thermodynamic quantity known as *entropy*. We will see more of this concept when we study quantifying information. At present we want to see how Landauer argued that information erasure causes an increase in the entropy of the environment and therefore a decrease in the energy of the system. His main point was that information was not something abstract, but was in fact the physical system used to represent it. In an illustrative example due to Szilard [68], a bit of information can be encoded in terms of the location of a molecule in the left or right of a partition in a transparent box (Figure 6.3). If we look at the box and find the molecule in the left partition then the system encodes a logical 0, and if it is on the right side then it encodes a logical 1. We can *write* one bit of information in this system by putting the molecule in the appropriate half.

One way to erase the information contained in the location of the molecule is to remove the partition and push the molecule to one end by compressing the "gas" with a piston. If we then replace the partition, the system reads 0 irrespective of what was encoded in it initially (Figure 6.4).

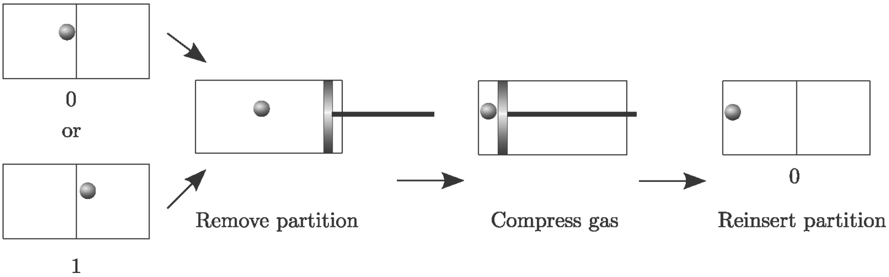


FIGURE 6.4: Erasing a bit of information.

Thermodynamics tells us how to calculate the work done in this process. The entropy of a thermodynamic system is related to the logarithm of the number of microscopic states available to the system. Since the molecule could be in one of two locations, the entropy associated with the single bit encoded in the box is given by $k_B \ln 2$, where k_B, the Boltzmann constant[1], is a proportionality factor. If the process of erasure is carried out at a constant temperature T then the energy dissipated in this process equals the work done in pushing the piston, which is $k_B T \ln 2$. This is the least amount of energy that is lost per bit when one performs an irreversible computation.

While there has been considerable debate in the literature regarding Landauer's principle, recently there appears to be experimental confirmation of the heat dissipated when a bit of information is erased. [12].

6.3.1 Classical reversible gates

In the early 1970s, this line of thinking prompted Bennett to come up with ways to beat the Landauer limit: by introducing reversible computation. The gates we have studied so far, such as AND and XOR, are intrinsically irreversible since they are two-one functions. An $n \to m$ function can, however, be implemented reversibly if it is embedded in a reversible $n + m \to m + n$ function.

The additional m inputs take certain fixed values, and are referred to as **ancilla** bits, while the extra n outputs are ignored. These are sometimes referred to as **garbage** bits.

$$\begin{array}{ccc} (x, 0) & \longmapsto & \big(f(x), g(x)\big). \\ \swarrow \quad \downarrow & & \downarrow \quad \searrow \\ \text{input} \quad \text{ancilla} & & \text{output} \quad \text{garbage} \end{array} \tag{6.13}$$

The advantage of reversibility is that the entire process can be run in reverse after storing (copying) the answers, so that all the bits are returned to their original states. The garbage is thus effectively recycled! The circuit diagram for such a reversible implementation is given in Figure 6.5.

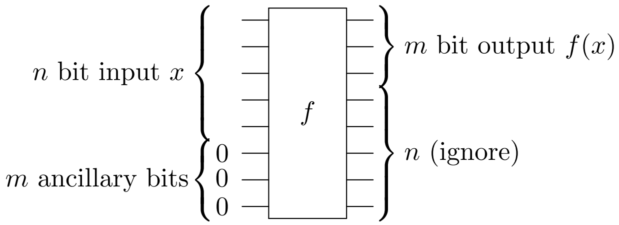


FIGURE 6.5: Reversible implementation of an irreversible function

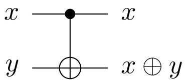
The function may be reversible only if the circuit to compute it is built out of reversible gates. NOT is a reversible 1-bit gate. A reversible 2-bit gate is the CNOT or controlled-NOT gate. This classic gate acts as NOT on the

[1] $k_B = 1.38 \times 10^{-23}$J/K. This constant appears in the relationship between energy and temperature at the level of particle constituents of a thermodynamic system.

second (target) input bit if the first (control) bit is set to 1; otherwise it leaves it unchanged. The truth table and circuit representation is:

CNOT:

x	y	x'	y'
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0



(6.14)

Here the top bit is the control bit. The filled circle on the connecting wire between the two bits represents control by the value 1. The lower bit is the target bit.

Exercise 6.3. Check that the matrix representation for the CNOT gate is

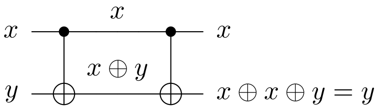
$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}.$$

Now the CNOT gate is a reversible implementation of the XOR gate. You can see that the second output y' represents the XOR of the inputs. So if we ignore the first output, we have here a reversible XOR gate:

$$XOR : (x, y) \mapsto (x, x \oplus y). \tag{6.15}$$

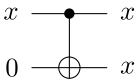
This is reflected in the circuit symbol for CNOT, where the target bit is shown with an $\oplus$ symbol acting on it, controlled by the first bit.

It is easy to see that this gate is the inverse of itself: if a second CNOT acts on the outputs of one CNOT, we get back the inputs to the first CNOT. (Note however that a reversible gate is not necessarily its self-inverse.)



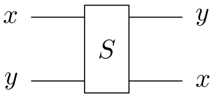
The CNOT gate can be used to reversibly embed several other useful gates such as the COPY gate and the SWAP gate:

$COPY : (x, 0) \mapsto (x, x)$

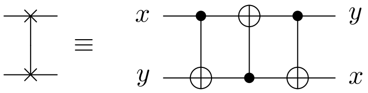


(6.16)

$SWAP : (x, y) \mapsto (y, x)$



(6.17)



where $(x, y) \xrightarrow{CNOT_{12}} (x, x \oplus y) \xrightarrow{CNOT_{21}} (y, x \oplus y) \xrightarrow{CNOT_{12}} (y, x)$.

6.3.2 Universal reversible gates

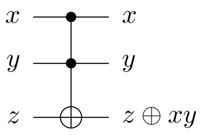
The classical universal sets obtained earlier, including AND, NOR, etc. are not reversible. The question now is whether our pet CNOT is universal. One way to see why it is not, is given by Preskill [57]. It turns out that the CNOT gate, as in fact, all 2-bit reversible gates, is an **affine** transformation. Any 2-bit gate whose output is a permutation of the input bits is of the form

$$\begin{bmatrix} x \\ y \end{bmatrix} \mapsto M \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} a \\ b \end{bmatrix} \tag{6.18}$$

where M is an invertible matrix and a and b are constants. There are invertible functions that are non-affine, especially for $n > 3$. Therefore, 2-bit gates are insufficient to generate such functions. Research has shown that certain conditional 3-bit gates are in fact universal. The most important for these are:

- **gate**: T is a doubly controlled NOT gate. Two control bits have to be set to 1 for NOT to act on the third bit. Else nothing changes. The truth table and circuit representation are as follows:

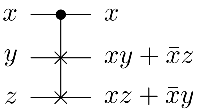
x	y	z	x'	y'	z'
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	0	1
1	1	0	1	1	1
1	1	1	1	1	0



(6.19)

- **Fredkin gate**: F is a controlled swap gate. If the control bit is set, then the other two bits are swapped.

x	y	z	x'	y'	z'
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	1	0
1	1	0	1	0	1
1	1	1	1	1	1



(6.20)

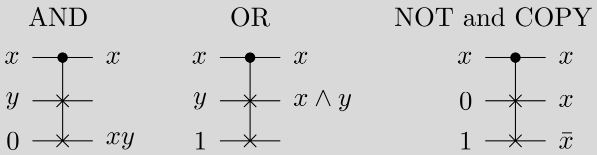
Exercise 6.4. Find out the matrix representations for the T and F gates.

Box 6.1: Billiard Ball Reversible Computer

The Fredkin gate has an interesting origin. It arose out of a mechanical model for reversible computation based on elastic collisions of a system of billiard balls and reflecting walls in a frictionless environment, proposed in 1982 by Fredkin and Toffoli [35]. A ball appearing at a port represents a logical 1 at that port, while the absence of a ball at a port represents a logical 0. The movement is restricted to a grid with unit distance, and the balls have radius $1/\sqrt{2}$ to capture discrete time steps. Inside the "computer", a billiard ball shot forward in a direction 45° up could collide with another ball or with horizontal reflecting walls, so that it always stays on the grid. At the output ports one obtains a readout of the process based on which ports are occupied and which are not. A series of well-placed reflectors would achieve a circuit built out of Fredkin gates. Since the collisions of the balls with the reflectors are nearly perfectly elastic, no energy is lost and we have an energy-conserving implementation of a reversible computation. A further property of the Fredkin gate, reflected by the billiard ball model, is that it is *conservative*, that is, the number of 1's in the input is preserved in the output. This just translates into no ball being lost in the computer. Conservativeness is also a concern of physical implementations of computation.

One way to see how such a 3-bit gate may be universal is to show how to implement the (irreversible) universal set {AND, NOT, OR, COPY} using only this gate.

Example 6.3.1. The universality of the Fredkin gate can be demonstrated by using it to implement the four universal logic gates:



For the output of the OR gate, we have used

$$x + \bar{x}y = x + (1 - x)y = x + y - xy = x \wedge y.$$

Also note how the required output appears at one port and the other ports are ignored. This is a common feature in implementing irreversible gates embedded in bigger, reversible ones.

Exercise 6.5. Show how {AND, NOT, OR, COPY} can be implemented by Toffoli gates alone.

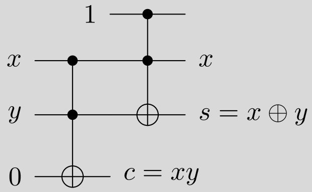
Thus, these classical universal gates can implement any function, provided some of the inputs are chosen to take fixed values, and some of the outputs are ignored, as in Figure 6.5.

Example 6.3.2. A reversible half-adder:

Let's see how to build a simple reversible circuit, for example a 1-bit adder, using Toffoli gates alone. The function we need must calculate the sum which is addition mod 2 of the inputs: $s = x \oplus y$, and the carry which is the AND of the inputs: $c = xy$.

$$f_{\text{add}}(x, y \ldots) = (x, y, x \oplus y, xy).$$

Check that the following circuit does what we need:



Exercise 6.6. Construct a half-adder using Fredkin gates alone.

6.4 Resources and Computational Complexity

We now come to the issues that make computer scientists look to the quantum paradigm for answering some of their questions on efficient algorithms. Efficiency is quantified in terms of how the resources used for the computation scale with the number of input bits n. Typically, polynomial scaling is termed "efficient" while exponential scaling is not. Resources are typically time, space and energy, though the last one is less a theoretical concern than for the physical implementation.

When analyzing the efficiency of an algorithm, it is desirable to factor out dependencies on the kind of computer the algorithm may be implemented on. The resulting features are to be intrinsic to the mathematical problem itself, and are defined in terms of how they scale as a function of the input size, rather than in absolute terms. These behaviors are termed **complexity**.

Time complexity is the most commonly considered aspect of efficiency of algorithms, and can be quantified by the number of elementary steps, such as the addition of two numbers, in the execution of an algorithm. Space complexity can be quantified by the amount of memory to be allocated to the execution of the algorithm. While the actual complexity of a problem depends on the particular algorithm used and also the size of the input, we try to generalize the concept by considering the asymptotic behavior, as the input size becomes very large.

Computational complexity is often quantified in three different ways. The way in which an algorithm scales as n is expressed in the following ways for large n:

1. $\mathcal{O}(g(n))$ (big oh): which specifies that the function $g(n)$ is the upper bound on the behavior of a resource;

2. $\Omega(g(n))$ (big omega): specifies that the function $g(n)$ is the lower bound on the behavior of a resource;

3. $\Theta(g(n))$ (big theta): this is the strongest condition, when a given resource scales as both $\mathcal{O}(g(n))$ and $\Omega(g(n))$ with the same function $g(n)$.

The first type, $\mathcal{O}$ which gives the upper bound, is the most commonly used.

Example 6.4.1. Let's look at the time complexity of simple arithmetic operations.

- Addition of two n-bit integers takes exactly n steps and has complexity $\Theta(n)$.

- Multiplication of two n-bit integers by the usual brute force method takes $n - 1$ additions and at most n carries. Thus the complexity is $\mathcal{O}(n^2)$.

- Matrix multiplication of two $n \times n$ matrices takes n multiplications and n additions, and therefore is of complexity $\mathcal{O}(n \times n^2) = \mathcal{O}(n^3)$. If the matrices are not square, but $m \times n$ and $n \times l$ then the complexity is $\mathcal{O}(mnl)$.

Box 6.2: Complexity Classes

There exists a plethora of complexity classes in this vast and deep subject and we list some of the more important ones here. These complexity classes assume a Turing model for the computer.

- **P** (Polynomial time): This class contains problems that are solvable in polynomial time, that is they are of $\mathcal{O}(n^k)$ for some k, on a deterministic Turing machine.

- **NP** (Non-deterministic polynomial time): this is the class of decision problem (with only "yes" or "no" answers) for which, given a solution, it can be verified in polynomial time in a non-deterministic Turing machine model. It is yet an unsolved problem as to whether an NP problem can be solved in polynomial time. Examples include integer factorization and discrete logarithm.

- **coNP** consists of decision problems whose complement is in NP.

- **NP-complete (NPC)** is the class of problems containing the hardest problem in NP. This class includes problems which may be outside NP. Examples are the Knapsack problem, the traveling salesman problem, Boolean satisfiability problem.

In some situations, especially in the quantum algorithms we are going to study in this book, we talk of the "query complexity" of an algorithm. Here, the algorithm is reduced to a series of binary answers to a query made to a function evaluator looked upon as a black box, whose functioning is unknown to us. This is calculated as the number of times the black box has to be queried to get to the solution. Of course, if the black box is replaced by a "white box": the details of the circuit used to implement the function, then we can relate the query complexity to the actual computational complexity of the entire process.

Part III

Quantum Computation

Chapter 7

Quantum Gates and Circuits

We are now ready to see how computing with qubits can be done. In this book, we will mainly use the circuit model for computation which was first introduced by Deutsch [25]. We will represent by quantum "wires," the qubits upon which manipulations. The length of the wire is to be interpreted as the time axis. Manipulations on qubits can be done using basic unitary operators that are the equivalents of logic "gates." An algorithm, or a complete set of steps for achieving a processing task, is a combination of wires and gates representing a quantum circuit. This circuit must be thought of as a time sequence of events with every wire a way of representing qubit states, and with gates representing processing of those states.

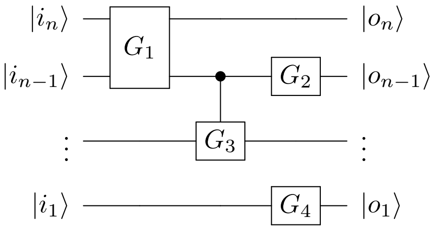
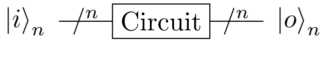


FIGURE 7.1: Illustrating a quantum circuit with n qubits.

This notation is based on one by Richard Feynman, with the convention that time flows from left to right.

Sometimes, an n-qubit state is represented by a wire with a $/^n$ decoration on it, that is referred to as a **register**.



In practice the circuit is effectively a unitary operator acting on the input qubits. A few major differences between classical circuits and quantum ones are:

- Quantum circuits never contain loops or feedbacks: they are *acyclic*
- Quantum wires are never fanned out: since arbitrary quantum states cannot be cloned

- Though the action of a circuit can be analyzed using classical states, the effect on superpositions is what gives it true quantum power.

The fact that quantum evolution is unitary results in quantum gates (and circuits) being reversible. This means that any manipulation of quantum information can be undone, unless an irreversible process such as measurement or decoherence happens on the system. This, and the peculiar features of qubits discussed in Chapter 4, makes for startling differences in the way we must think about quantum algorithms.

Mathematically a gate can be represented as a matrix. Classical reversible gates can have only ones and zeros as elements: reversibility implies that they can only perform a permutation of the inputs. For example, a reversible XOR gate is given by y' output in the truth table of what has sometimes been called a Feynman gate:

x	y	x'	y'
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0

$$\Longrightarrow \begin{array}{c c} & \begin{array}{cccc} 00 & 01 & 10 & 11 \end{array} \\ \begin{array}{c} 00 \\ 01 \\ 10 \\ 11 \end{array} & \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \end{array} \tag{7.1}$$

Such a gate is also implementable as a quantum gate, but the most generic quantum gate is represented by a complex matrix.

7.1 Single Qubit Gates

Classically, there exists only one reversible single bit gate: the NOT gate which effects $0 \to 1, 1 \to 0$. However, any unitary operation on the qubits $|0\rangle$ and $|1\rangle$ is a valid single qubit gate. As we will see, such a gate can always be regarded as a linear combination of the Pauli gates X, iY, Z and the identity.

In circuit notation, a gate G that acts on state $|i\rangle$ to produce state $|o\rangle$ is represented as

$$|i\rangle \;-\boxed{G}-\; |o\rangle$$

The matrix representation of G is found by computing its action on the computational basis states:

$$G_{ij} = \langle i|G|j\rangle \tag{7.2}$$

The full power of the quantum gate emerges when it acts on superposition states. Consider for example the action of NOT, defined in the computational basis by

$$\begin{aligned} X|0\rangle &= \ |1\rangle \\ X|1\rangle &= \ |0\rangle \end{aligned} \ ; \qquad X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} = \sigma_x \tag{7.3}$$

When X acts on a generic quantum state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ we get $X|\psi\rangle = \alpha|1\rangle + \beta|0\rangle$. This represents interchanged probabilities of the state being in $|0\rangle$ or $|1\rangle$.

Other useful quantum single-qubit gates, that have no classical analogue, are described below.

1. Phase Flip (Z) gate:

$$\begin{aligned} Z|0\rangle &= \ |0\rangle \\ Z|1\rangle &= \ -|1\rangle \end{aligned} \ ; \qquad Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} = \sigma_z \tag{7.4}$$

This gate gives the state $|1\rangle$ a negative sign, an operation that is meaningless in classical logic, but is relevant when it acts on superposition states of a qubit. For instance, the state $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ changes to the orthogonal state $\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$.

2. Hadamard (H) gate:

$$\begin{aligned} H|0\rangle &= \ \tfrac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right); \\ H|1\rangle &= \ \tfrac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right); \end{aligned} \qquad H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \frac{1}{\sqrt{2}}\left(\sigma_x + \sigma_z\right) \tag{7.5}$$

This is an invaluable gate in quantum information processing: it produces equal superpositions of the basis states. Its action can be expressed algebraically as

$$H|x\rangle = \frac{1}{\sqrt{2}}(|x\rangle + (-1)^x|\bar{x}\rangle) = \frac{1}{\sqrt{2}} \sum_{y=0,1} (-1)^{xy}|y\rangle. \tag{7.6}$$

3. Phase (Φ) gate: $$\begin{aligned} \Phi|0\rangle &= \ |0\rangle; \\ \Phi|1\rangle &= \ e^{i\varphi}|1\rangle \end{aligned} \ ; \qquad \Phi = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\varphi} \end{bmatrix} \tag{7.7}$$

Exercise 7.1. Show that the Z, H, and Φ matrices are all unitary.

Exercise 7.2. Calculate the output of each of these gates when the input is a general qubit state $\alpha|0\rangle + \beta|1\rangle$.

Exercise 7.3. What is the action of the Pauli Y gate?

It is useful to visualize the action of single qubit gates by looking at their action on the Bloch sphere. A gate must take any point on the Bloch sphere to another, and can be a rotation about an arbitrary axis through the center of the Bloch sphere. Inversions about the center are also allowed.

Example 7.1.1. To see the effect of the Pauli X matrix on a qubit state on the Bloch sphere,

$$\begin{aligned} X\left(\cos\frac{\theta}{2}|0\rangle + e^{i\phi}\sin\frac{\theta}{2}|1\rangle\right) &= \cos\frac{\theta}{2}|1\rangle + e^{i\phi}\sin\frac{\theta}{2}|0\rangle \\ &= e^{i\phi}\left[\cos\left(\frac{\pi}{2}-\frac{\theta}{2}\right)|0\rangle + e^{-i\phi}\sin\left(\frac{\pi}{2}-\frac{\theta}{2}\right)|1\rangle\right] \end{aligned} \tag{7.8}$$

This is a state for which $\theta \to \pi - \theta$ and $\phi \to -\phi$. The transformation is illustrated in Figure 7.2.

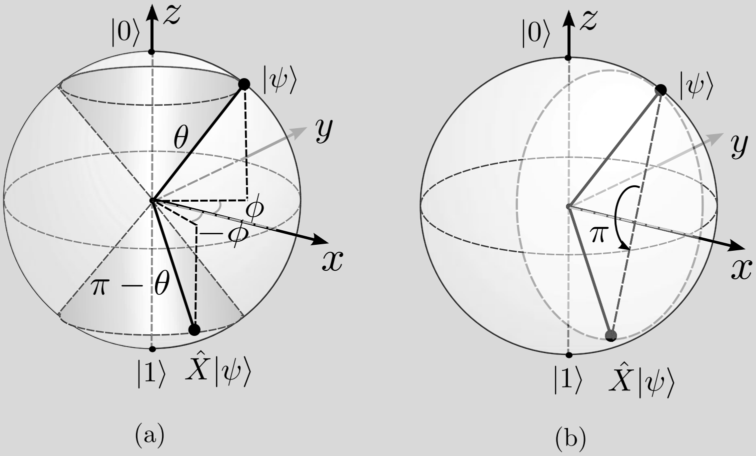


FIGURE 7.2: Action of $\hat{X}$ on the Bloch sphere.(a) The θ and $\pi-\theta$ cones are indicated to show you how the transformation works. (b) The result is equivalent to a rotation about $\hat{\boldsymbol{x}}$ by π.

Exercise 7.4. Show that the Pauli gates Y and Z gates rotate a state on the Bloch sphere by π about the $\hat{\boldsymbol{y}}$ and $\hat{\boldsymbol{z}}$ axes, respectively.

Exercise 7.5. The effect of the H gate on the Bloch sphere can also be regarded as a rotation by π about some axis. Find that axis.

Exercise 7.6. What is the effect of the phase gate Φ on a state located at (θ, ϕ) on the Bloch sphere?

A general rotation can always be constructed as combinations of rotations about the $\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}$, and $\hat{\boldsymbol{z}}$ axes. Hence a very useful set of gates is the rotation gates, expressed as functions of the Pauli matrices as follows:

$$R_x(\theta) \equiv e^{-i\theta\sigma_x/2} = \begin{bmatrix} \cos\frac{\theta}{2} & -i\sin\frac{\theta}{2} \\ -i\sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{bmatrix} \tag{7.9a}$$

$$R_y(\theta) \equiv e^{-i\theta\sigma_y/2} = \begin{bmatrix} \cos\frac{\theta}{2} & -\sin\frac{\theta}{2} \\ \sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{bmatrix} \tag{7.9b}$$

$$R_z(\theta) \equiv e^{-i\theta\sigma_z/2} = \begin{bmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{bmatrix} \tag{7.9c}$$

Exercise 7.7. Show by using the series expansion of e^x that if A is a matrix such that $A^2 = \mathbb{1}$ then $e^{iA\theta} = \cos(\theta)\mathbb{1} + i\sin(\theta)A$.

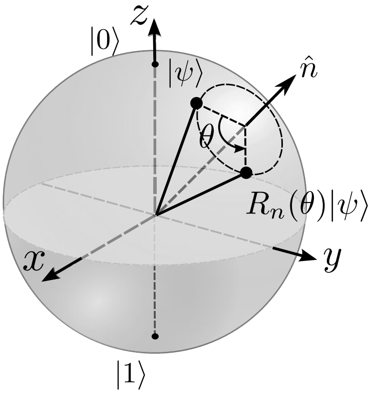


FIGURE 7.3: Rotation of a qubit by $R_n(\theta)$ on the Bloch sphere.

You can now see that a rotation about an axis $\hat{\boldsymbol{n}} = n_x\hat{\boldsymbol{i}} + n_y\hat{\boldsymbol{j}} + n_y\hat{\boldsymbol{k}}$ by an angle θ is given by

$$R_{\hat{\boldsymbol{n}}}(\theta) = e^{-i\theta\hat{\boldsymbol{n}}\cdot\vec{\boldsymbol{\sigma}}/2} = \cos\left(\frac{\theta}{2}\right)\mathbb{1} - i\sin\left(\frac{\theta}{2}\right)(n_x\sigma_x + n_y\sigma_y + n_z\sigma_z)\,. \tag{7.10}$$

The action of this gate is illustrated in Figure 7.3.

Exercise 7.8. Verify that gate $R_{\hat{\boldsymbol{n}}}(\theta)$ takes a state with Bloch vector $\hat{\boldsymbol{a}}$ to one rotated by θ about the $\hat{\boldsymbol{n}}$ axis.

Box 7.1: Useful Representations of Single Qubit Gates

The Pauli matrices, along with the 2×2 identity matrix, are said to form a basis for the space of 2×2 matrices. So any single qubit gate A can be expressed as a linear combination

$$A = a_0 \mathbb{1} + \vec{\boldsymbol{a}} \cdot \vec{\boldsymbol{\sigma}}.$$

1. Since A is unitary, it can be expressed up to an overall phase as

   $$U = u_0 \mathbb{1} + i \vec{\boldsymbol{u}} \cdot \vec{\boldsymbol{\sigma}}, \tag{7.11}$$

   for *real* u_0, u_1, u_2, u_3 s.t. $u_0^2 + \vec{\boldsymbol{u}} \cdot \vec{\boldsymbol{u}} = 1$.

2. This can be re-expressed as

   $$U = e^{i\alpha} e^{i\beta \hat{\boldsymbol{n}} \cdot \vec{\boldsymbol{\sigma}}}, \tag{7.12}$$

   where α is a phase, $\hat{\boldsymbol{n}}$ is a unit vector parallel to $\vec{\boldsymbol{u}}$ and β is an angle, which turns out to be half the angle of rotation of the initial state about the axis $\hat{\boldsymbol{n}}$.

Successive action of gates

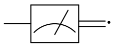
Two successive operations are two unitary gates, say A and B, acting one after another. Algebraically, we represent the resultant by the action of the usual matrix product of the two gates:

$$|\psi\rangle \xrightarrow{A} A|\psi\rangle \xrightarrow{B} BA|\psi\rangle. \tag{7.13}$$

Note that the order of the gates is important. Operators do not in general commute. The circuit representation of this process is like a time sequence, and the order of gates is obvious:

$$|\psi\rangle \; -\boxed{A}-\boxed{B}-$$

7.1.1 Measurement gate

At the end of a computation we need to measure the output in order to read out the result of the computation. This leads to obtaining classical information (in bits) out of the quantum system. One sets up an experiment that measures an appropriate physical quantity to give one of its eigenvalues as the result (recall Section 3.3 and the nature of measurements in quantum mechanics). We denote this process generically by a *measurement gate* . The double line for the output state is to emphasize that it is a classical state. By default the measurement is assumed to be in the computational basis. The state just prior to measurement encodes the probabilities of its collapsing to $|0\rangle$ or $|1\rangle$.

7.2 Multi-Qubit Gates

Two qubits together can be represented as 4-column vectors in Hilbert space. The most general 2-qubit gate is therefore a 4×4 unitary. An operation on two qubits that acts independently on each of the two can be expressed as a direct product of two single-qubit operations as defined in Equation 3.31:

$$O = O_1 \otimes O_2.$$

For example, the 2-qubit H gate is represented by the action

$$H^{\otimes 2}|x\rangle|y\rangle = H|x\rangle \otimes H|y\rangle, \tag{7.14}$$

with matrix representation

$$\frac{1}{2}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}. \tag{7.15}$$

You need to distinguish between the different possibilities shown in Figure 7.4. The circuit diagrams for these gates will clarify the difference.

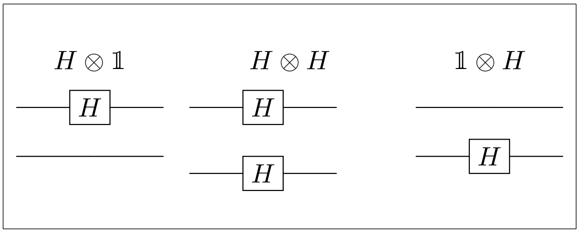


FIGURE 7.4: H gates acting in different ways on two qubits.

These sort of gates can easily be generalized to any dimensions.

Exercise 7.9. Construct the matrix representations for the operators shown in Figure 7.4.

Exercise 7.10. Find the matrix representing $X \otimes Z$.

The interesting thing about multi-qubit gates is that in general, they would not act independently on the individual qubits, but entangle them. This is the hallmark of quantum information processing that gives the most crucial advantage over classical processing. For example, consider the most famous 2-qubit gate, the controlled-NOT or CNOT gate whose classical version we saw in Chapter 6. This gate flips the target qubit when the control qubit is set to 1. The truth table of the CNOT is used to define the action of the quantum gate on the computational basis states:

$$U_{\text{CNOT}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \equiv \begin{bmatrix} \mathbb{1} & 0 \\ 0 & X \end{bmatrix} \tag{7.16}$$

Notice that the truth table for the second output corresponds to the well-known XOR operation on the inputs. The operation is, however, completely reversible. We denote the action of this gate by

$$U_{\text{CNOT}}|x\rangle|y\rangle = |x\rangle|x \oplus y\rangle. \tag{7.17}$$

Note that when we use letters x and y to label quantum states, they refer to the computational basis states. This gate is represented by the circuit of Figure 7.5. An important caveat here: though the control qubit seems to come out of the gate unchanged when it is in a computational basis state, the output will in general be entangled with the state of the target qubit, as we will see in the next example.

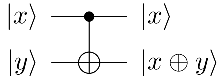


FIGURE 7.5: CNOT gate.

Example 7.2.1. As an illustration of how a controlled gate acts on superposition states, consider

$$\begin{aligned} CNOT(\alpha|0\rangle + \beta|1\rangle)|0\rangle &= CNOT(\alpha|00\rangle + \beta|10\rangle) \\ &= \alpha|00\rangle + \beta|11\rangle \end{aligned} \tag{7.18}$$

which is an entangled state. Figure 7.6 gives the circuit for this process.

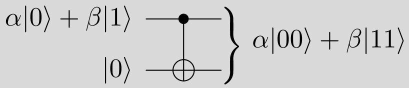

$\alpha|0\rangle + \beta|1\rangle$
$|0\rangle$
$\alpha|00\rangle + \beta|11\rangle$

FIGURE 7.6: CNOT producing entanglement.

This example also illustrates the No-cloning theorem of Chapter 4. The CNOT gate appears as a cloner if the target qubit is $|0\rangle$:

$$U_{\text{CNOT}}|x\rangle|0\rangle = |x\rangle|x\rangle. \tag{7.19}$$

However, this is true iff $|x\rangle$ is a computational basis state. If the control qubit is a generic quantum state $|\psi\rangle$, the output of this gate is an *entangled* state. If our gate were a cloner, then the output ought to have been $|\psi\rangle \otimes |\psi\rangle$, which is a separable state.

The notion of a conditional or controlled gate can be extended to any unitary single-qubit operation U by defining

$$U_{CU}|x\rangle|y\rangle = |x\rangle U^x|y\rangle \tag{7.20}$$

The notation makes it obvious that the operator U acts on the target qubit $|y\rangle$ only if the control qubit is set to 1. Figure 7.7 shows the circuit representation for this action.

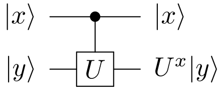


FIGURE 7.7: Circuit representing a controlled-U gate.

The matrix representation of such a gate is

$$U_{CU} = \begin{bmatrix} \mathbb{1} & 0 \\ 0 & U \end{bmatrix}. \tag{7.21}$$

You can prove that U_{CU} is unitary if U is.

One can use either of the input qubits as the control or the target. We will use the notation C_{ij} to denote the i^{th} bit as the control bit and the j^{th} bit as the target.

Exercise 7.11. Show that $(H \otimes H)C_{12}(H \otimes H) = C_{21}$, i.e., if you change basis from computational basis to the X basis $\{|+\rangle, |-\rangle\}$, then the control and target bits get interchanged. The circuit for the problem looks like Figure 7.8.

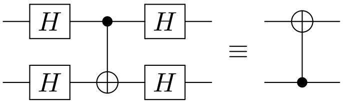


FIGURE 7.8: CNOT with second qubit as control and first as target.

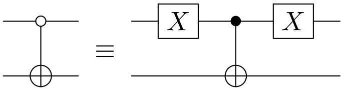


FIGURE 7.9: A 0-controlled gate.

The control action can be conditioned on the control bit set to 0 instead of 1. Such a gate is represented in Figure 7.9.

For more than one qubit, a variety of control possibilities are illustrated in Figure 7.10.

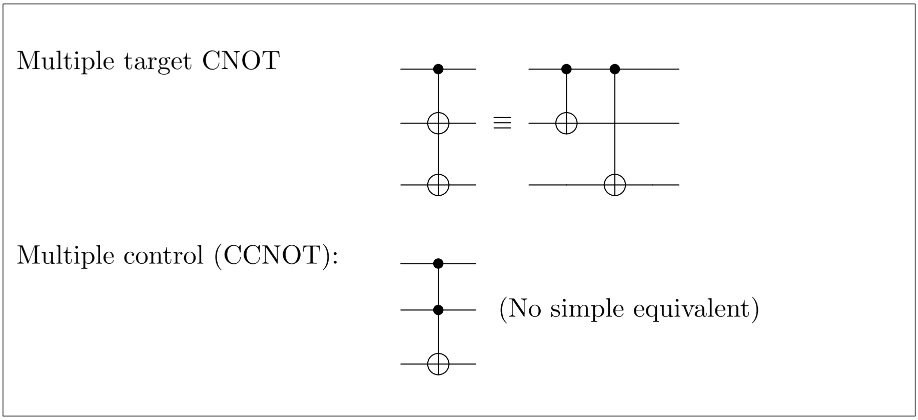


FIGURE 7.10: Different control operations

Example 7.2.2. Creating Bell states

Prototype entangled states are the Bell states of Equation 4.10, and they can be produced using CNOT gates. For example,

$$|0\rangle \otimes |0\rangle \xrightarrow{H\otimes \mathbb{1}} \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |0\rangle \xrightarrow{C_{12}} \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle), \tag{7.22}$$

producing the first Bell state $|\beta_{00}\rangle$. It's easy to deduce that the general Bell state is produced by the simple circuit given in Figure 7.11:

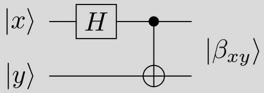


FIGURE 7.11: Circuit for preparing Bell States

Exercise 7.12. Verify that the operation depicted in circuit 7.11 is reversible.

Exercise 7.13. Verify that the Bell states can be written as

$$|\beta_{xy}\rangle = \frac{1}{\sqrt{2}}\left(|0y\rangle + (-1)^x|1\bar{y}\rangle\right).$$

The reverse of the circuit 7.11 can be used to convert the Bell basis to the computational one. Making a measurement after that can tell us which of the Bell states we started with. This is called a *Bell Measurement*, depicted in Figure 7.12.

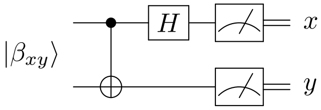


FIGURE 7.12: Circuit for Bell measurement.

Example 7.2.3. Let us analyze the output of the circuit shown in Figure 7.13. $|\psi\rangle$ is a generic unknown qubit $\alpha|0\rangle+\beta|1\rangle$. A Bell measurement is performed on this qubit and one of an entangled pair prepared in the state $|\beta_{00}\rangle$ (Equation 4.10).

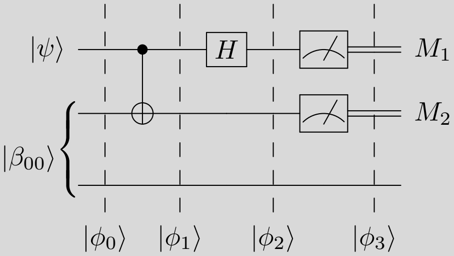


FIGURE 7.13: Bell measurement on part of an entangled state.

We will algebraically analyze the output at each stage of the circuit:

$$\begin{aligned}
|\phi_0\rangle &= |\psi\rangle \otimes |\beta_{00}\rangle = (\alpha|0\rangle + \beta|1\rangle) \otimes \frac{1}{\sqrt{2}}\left(|00\rangle + |11\rangle\right) \\
&= \frac{1}{\sqrt{2}}\left(\alpha|000\rangle + \alpha|011\rangle + \beta|100\rangle + \beta|111\rangle\right). \\
|\phi_1\rangle &= C_{12} \otimes \mathbb{1}|\phi_0\rangle = \frac{1}{\sqrt{2}}\left(\alpha|000\rangle + \alpha|011\rangle + \beta|110\rangle + \beta|101\rangle\right). \\
|\phi_2\rangle &= H \otimes \mathbb{1} \otimes \mathbb{1}|\phi_1\rangle \\
&= \frac{\alpha}{\sqrt{2}}\left[\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)(|00\rangle + |11\rangle)\right] \\
&\quad + \frac{\beta}{\sqrt{2}}\left[\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)(|10\rangle + |01\rangle)\right].
\end{aligned}$$

Now $|\phi_3\rangle$ is a single-qubit state of the last wire, obtained after measuring the first two qubits of $|\phi_2\rangle$. So let's regroup the terms in $|\phi_2\rangle$ separating out the states of the first two qubits from the third:

$$\begin{aligned} |\phi_2\rangle &= \frac{1}{2}\left[\alpha(|000\rangle + |100\rangle + |011\rangle + |111\rangle)\right. \\ &\quad \left. +\beta\left(|010\rangle + |001\rangle - |110\rangle - |101\rangle\right)\right] \\ &= \frac{1}{\sqrt{2}}|00\rangle\left[\frac{1}{\sqrt{2}}(\alpha|0\rangle + \beta|1\rangle)\right] \\ &+ \frac{1}{\sqrt{2}}|01\rangle\left[\frac{1}{\sqrt{2}}(\alpha|1\rangle + \beta|0\rangle)\right] \\ &+ \frac{1}{\sqrt{2}}|10\rangle\left[\frac{1}{\sqrt{2}}(\alpha|0\rangle - \beta|1\rangle)\right] \\ &+ \frac{1}{\sqrt{2}}|11\rangle\left[\frac{1}{\sqrt{2}}(\alpha|1\rangle - \beta|0\rangle)\right] \end{aligned}$$

When the first two qubits are measured, $|\phi_2\rangle$ collapses to the state corresponding to the output. This leaves the third qubit in a corresponding state, that is closely related to $|\psi\rangle$as tabulated in Table 7.1.

TABLE 7.1: Resulting state after measurement.

Measurement result	$\|\psi_3\rangle$
00	$\|\psi\rangle$
01	$X\|\psi\rangle$
10	$Z\|\psi\rangle$
11	$XZ\|\psi\rangle$

The idea behind this circuit is quantum state teleportation, which will be further discussed in Section 9.1.1.

Example 7.2.4. Measuring an operator

Consider a unitary operator $\hat{U}$ that can be used as a quantum gate. If $\hat{U}$ happens to be an observable as well, then it must be Hermitian. So its eigenvalues must be ±1. The Pauli operators are examples of such operators. Now we'll show that the circuit in Figure 7.14 effects a measurement of $\hat{U}$ on the state $|\psi\rangle$ input in the lower register.

Remember, this means that at the end of the circuit, the meter reads 0 or 1 corresponding to the eigenvalues $+1$ or -1, and the state on the bottom wire must be the corresponding eigenstate $|u_+\rangle$ or $|u_-\rangle$ of $\hat{U}$.

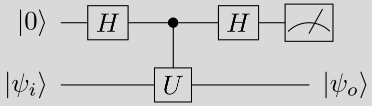


FIGURE 7.14: Circuit for measuring an operator

We have

$$\hat{U}|u_+\rangle = |u_+\rangle, \qquad \hat{U}|u_-\rangle = -|u_-\rangle.$$

We can expand the initial state in the U-basis:

$$|\psi_i\rangle = a|u_+\rangle + b|u_-\rangle.$$

Working through the circuit,

$$\begin{aligned}
|0\rangle \otimes |\psi_i\rangle \quad &\xrightarrow{H_1} \quad \frac{1}{\sqrt{2}}\big(|0\rangle + |1\rangle\big) \otimes \big(a|u_+\rangle + b|u_-\rangle\big) \\
&\qquad\; = \frac{1}{\sqrt{2}}\big(a|0\rangle|u_+\rangle + a|1\rangle|u_+\rangle + b|0\rangle|u_-\rangle + b|1\rangle|u_-\rangle\big) \\
&\xrightarrow{CU_{12}} \quad \frac{1}{\sqrt{2}}\big(a|0\rangle|u_+\rangle + a|1\rangle\hat{U}|u_+\rangle + b|0\rangle|u_-\rangle + b|1\rangle\hat{U}|u_-\rangle\big) \\
&\qquad\; = a\frac{\big(|0\rangle + |1\rangle\big)}{\sqrt{2}}|u_+\rangle + b\frac{\big(|0\rangle - |1\rangle\big)}{\sqrt{2}}|u_-\rangle.
\end{aligned}$$

On measuring the first qubit, in the X basis, we get 0 with probability $|a|^2$ and 1 with probability $|b|^2$ with the second qubit left in the corresponding eigenstate of $\hat{U}$. The circuit thus implements a measurement of the observable U. If the input state were an exact eigenstate of U then the corresponding eigenvalue is measured with probability 1.

7.3 Quantum Function Evaluation

We've taken the circuit analogy for quantum computation up to gates. Can we go further? Can we identify a set of universal gates, as we did for classical computation?

Since a computation is essentially the evaluation of a function of the inputs, let's first fix what we mean by a quantum function evaluation. Consider a function $f : \{0,1\}^n \mapsto \{0,1\}^m$ that takes an n-bit input x and produces an m-bit output $f(x)$. A reversible implementation of this function would have an $n+m$-bit input and the same number of bits in the output. We will use this to define the unitary operator implementing $f(x)$.

Definition 7.1. A quantum function evaluator *is a unitary operator* U_f*, for* $f : \{0,1\}^n \mapsto \{0,1\}^m$*, such that*

$$U_f|x\rangle|y\rangle = |x\rangle|f(x) \oplus y\rangle. \tag{7.23}$$

This is essentially an f-controlled XOR gate (which is like an f-controlled NOT gate if $m = 1$), expressed in the circuit of Figure 7.15.

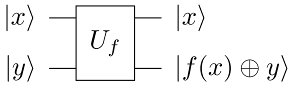


FIGURE 7.15: Quantum function evaluator.

Here, $|x\rangle$ is an n-qubit basis state while $|y\rangle$ is an m-qubit one. Note that U_f will be represented by an $n + m$ square matrix. If the input lower register $y = 0$, then the output on the register is just $f(x)$.

Exercise 7.14. Show that U_f as defined in Equation 7.23 is unitary and therefore reversible.

The important feature of a unitary transformation is not only that it admits an inverse, but also that it is linear. So it acts on superpositions thus:

$$U_f\left(c_1|x_1\rangle + c_2|x_2\rangle\right)|y\rangle = c_1 U_f\left(|x_1\rangle|y\rangle\right) + c_2 U_f\left(|x_2\rangle|y\rangle\right). \tag{7.24}$$

For instance, if the input is the uniform superposition of two qubits, the linearity of U_f means that

$$U_f \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right)|0\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle|f(0)\rangle + |1\rangle|f(1)\rangle\right).$$

The output is an entangled superposition state of both registers, containing both $f(0)$ as well as $f(1)$. This generalizes to multiple qubits as well. A uniform superposition of n qubits is the normalized sum of all 2^n possible n-qubit basis states $|0\rangle, |1\rangle \ldots |2^n - 1\rangle$. So we have

$$U_f \frac{1}{\sqrt{2^n}}\left(\sum_{x=0}^{2^n-1} |x\rangle_n\right)|0\rangle_m = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle_n |f(x)\rangle_m \tag{7.25}$$

where the subscripts on the states indicate the dimensionality. The function has been evaluated in parallel on all inputs. This has been referred to as **quantum parallelism**. The catch is, however, that this superposition does not mean much to our classical minds, until we measure the output, upon which **one** of the answers is selected! We can never know all the $f(x)$'s at once, nor can we clone the output and hope to learn $f(x)$ by making repeated measurements of the output state.

Nevertheless, this feature is enormously useful in designing quantum algorithms. One has to additionally choose clever modifications of the output such that the state containing the answer occurs with high amplitude. We will see this in action in the next chapters.

7.4 Universal Quantum Gates

We now wish to push the circuit analogy further and explore the possibility of universal quantum gates. Let's start with single-qubit gates. We've seen that these are 2 × 2 unitary matrices, which take a point on the Bloch sphere to another. It is easy to see that there are infinitely many possible 1-qubit gates. These however cannot form a universal set since controlled operations cannot be implemented by taking direct products of 1-qubit gates. How do we implement controlled gates in general?

7.4.1 Controlled-U gate

Working toward a general construction for a controlled U gate for arbitrary U makes use of the following representation for U:

Theorem 7.1. *Any unitary* 2×2 *matrix can be decomposed as*

$$U = e^{i\theta}\ A\ \sigma_x\ B\ \sigma_x\ C, \quad s.t \qquad A\ B\ C = \mathbb{1}, \tag{7.26}$$

where $A, B,$ *and* C *are also unitary.*

Proof. The proof hinges on the fact that any unitary matrix implements a rotation on the Bloch sphere, up to an over-all phase factor $e^{i\theta}$. Suppose V is some unitary matrix. The matrix $V\sigma_x V^\dagger$ is also unitary, so that it can be represented (see Equation 7.11) as

$$V\sigma_x V^\dagger = a_0\mathbb{1} + \vec{\boldsymbol{a}} \cdot \vec{\boldsymbol{\sigma}}, \quad a_0^2 + \vec{\boldsymbol{a}} \cdot \vec{\boldsymbol{a}} = 1.$$

But $V\sigma_x V^\dagger$ is a similarity transformation of σ_x. So it must preserve its trace, which is zero. Therefore $a_0 = 0$ and

$$V\sigma_x V^\dagger = \hat{\boldsymbol{a}} \cdot \vec{\boldsymbol{\sigma}} \text{ for a real unit } \hat{\boldsymbol{a}}.$$

Note that $\sigma_x = \hat{\boldsymbol{x}} \cdot \vec{\boldsymbol{\sigma}}$. Then $V\sigma_x V^\dagger$ must be rotating $\hat{\boldsymbol{x}}$ to a new direction $\hat{\boldsymbol{a}}$. Similarly, another unitary W will achieve

$$W\sigma_x W^\dagger = \hat{\boldsymbol{b}} \cdot \vec{\boldsymbol{\sigma}} \text{ for a real unit } \hat{\boldsymbol{b}}.$$

Thus we have

$$\begin{aligned} V\sigma_x V^\dagger\ W\sigma_x W^\dagger &= (\hat{\boldsymbol{a}} \cdot \vec{\boldsymbol{\sigma}})\ (\hat{\boldsymbol{b}} \cdot \vec{\boldsymbol{\sigma}}) \\ &= \hat{\boldsymbol{a}} \cdot \hat{\boldsymbol{b}}\mathbb{1} + i\ \hat{\boldsymbol{a}} \times \hat{\boldsymbol{b}} \cdot \vec{\boldsymbol{\sigma}}. \end{aligned}$$

(Refer to Equation 3.34 you proved in one of the problems of Chapter 3.) We can now think of $\hat{\boldsymbol{a}}$ and $\hat{\boldsymbol{b}}$ as directions with an angle γ between them so that

$$\hat{\boldsymbol{a}} \cdot \hat{\boldsymbol{b}} = \cos\gamma, \qquad \hat{\boldsymbol{a}} \times \hat{\boldsymbol{b}} = \sin\gamma\hat{\boldsymbol{n}}, \text{ which is perpendicular to } \hat{\boldsymbol{a}} \text{ and } \hat{\boldsymbol{b}}.$$

Then we can construct

$$\begin{aligned} U &= e^{i\theta} V \sigma_x V^\dagger \; W \sigma_x W^\dagger \\ &= e^{i\theta} \left(\cos\gamma \mathbb{1} + i \sin\gamma \hat{\boldsymbol{n}} \cdot \vec{\boldsymbol{\sigma}} \right) \\ &= e^{i\theta} e^{i\gamma \hat{\boldsymbol{n}} \cdot \vec{\boldsymbol{\sigma}}}, \end{aligned}$$

which is a valid representation for a unitary operator! If we identify

$$V = A, \quad V^\dagger \; W = B \text{ and } W^\dagger = C,$$

then we have the requisite representation for U. □

We can implement C-$V\sigma_x V^\dagger \; W\sigma_x W^\dagger$ by the circuit of Figure (7.16).

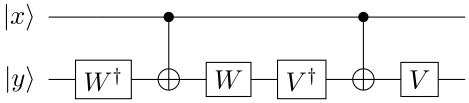


FIGURE 7.16: Circuit to evaluate C-U up to the phase factor

It is straightforward to see that when $x = 0$, the output is $VV^\dagger WW^\dagger y = y$, and when $x = 1$, the output is $V\sigma_x V^\dagger \; W\sigma_x W^\dagger y = Uy$ up to the phase. So this gives C-U up to the phase factor. We need to additionally implement the controlled phase C-Θ where

$$\Theta = \begin{bmatrix} e^{i\theta} & 0 \\ 0 & e^{i\theta}. \end{bmatrix}$$

Now check that

$$\text{C}{-}\Theta = \begin{bmatrix} \mathbb{1} & 0 \\ 0 & \Theta \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{bmatrix} \otimes \mathbb{1}.$$

We then have the implementation of the full C-U illustrated in Figure (7.17), that uses only CNOT gates and single-qubit gates.

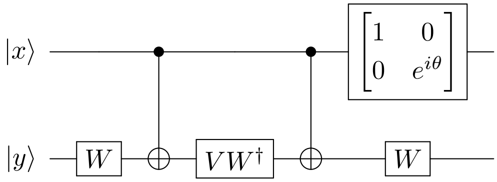


FIGURE 7.17: Implementation of controlled U gate.

Here, V and W are arbitrary unitaries. We are now half-way through in our quest for universal quantum gates, of which one set is given in the following theorem:

Theorem 7.2. ***Universal Quantum Gates:*** *the CNOT gate along with single-qubit gates is universal.*

How do we prove this? Now classically, the Toffoli gate, which is a C-C-NOT gate, is universal. We'll now show that given our construction for C-U gates, we can build doubly controlled C-C-U gates as follows. Consider a unitary Q such that $Q^2 = U$. Then we can build a C-C-U by the circuit in Figure 7.18. Let's work through this circuit algebraically to show that it works as expected:

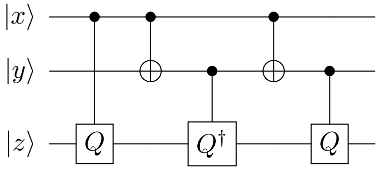


FIGURE 7.18: Implementation of C-C-U gate.

$$\begin{aligned}
|x\rangle|y\rangle|z\rangle &\rightarrow |x\rangle|y\rangle\, Q^x|z\rangle \\
&\rightarrow |x\rangle|x\oplus y\rangle\, Q^x|z\rangle \\
&\rightarrow |x\rangle|x\oplus y\rangle\, (Q^\dagger)^{x\oplus y}\, Q^x|z\rangle \\
&\rightarrow |x\rangle|y\rangle\, (Q^{-1})^{x\oplus y}\, Q^x|z\rangle \\
&\rightarrow |x\rangle|y\rangle\, Q^y\, Q^{-x\oplus y}\, Q^x|z\rangle
\end{aligned}$$

The power of Q that acts on $|z\rangle$ in the end is

$$y - (x\oplus y) + x = y - (x + y - 2xy) + x = 2xy.$$

So the effect of this circuit is

$$|z\rangle \rightarrow Q^{2xy}|z\rangle = U^{xy}|z\rangle,$$

which is exactly what we want. We can for instance construct a quantum Toffoli gate by using $Q^2 = X$. One such "square root of NOT" gate is

$$\sqrt{X} = \frac{1}{2}\begin{bmatrix} 1+i & 1-i \\ 1-i & 1+i \end{bmatrix}. \tag{7.27}$$

Example 7.4.1. A useful question to ask in designing circuits is how to minimize the number of basic gates required for a given implementation. In our construction for the C-C-U gate given above, we require 2 CNOTs plus 2 CNOTs for each C-Q gate, that is a total of 8 CNOTs. Can we be more frugal? Here is an example from Mermin [48] of a construction for a Toffoli gate using only 4 CNOT gates. Consider two unitaries A and B such that $A^2 = \mathbb{1} = B^2$. This means that

$$A = V^\dagger X V, \qquad B = W^\dagger X W.$$

Thus each C-A and C-B gate requires only one CNOT gate and two single-qubit gates.

You should be able to work out that the circuit of Figure 7.19 implements a doubly controlled $(BA)^2$ gate, up to a phase α.

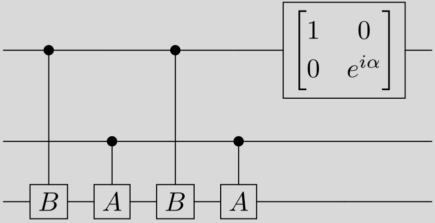


FIGURE 7.19: Efficient implementation of a Toffoli gate.

Now

$$\begin{aligned} AB &= V^\dagger X V W^\dagger X W = (\hat{\boldsymbol{a}} \cdot \vec{\boldsymbol{\sigma}})(\hat{\boldsymbol{b}} \cdot \vec{\boldsymbol{\sigma}}) \\ &= \hat{\boldsymbol{a}} \cdot \hat{\boldsymbol{b}} \mathbb{1} + i(\hat{\boldsymbol{a}} \times \hat{\boldsymbol{b}}) \cdot \vec{\boldsymbol{\sigma}}. \end{aligned}$$

If we choose the angle between $\hat{\boldsymbol{a}}$ and $\hat{\boldsymbol{b}}$ to be $\pi/4$, and also let $\hat{\boldsymbol{a}} \times \hat{\boldsymbol{b}}$ point along $\hat{\boldsymbol{x}}$, then we have

$$\begin{aligned} AB &= \cos\frac{\pi}{4} + i \sin\frac{\pi}{4}\sigma_x = \frac{1}{\sqrt{2}}(\mathbb{1} + iX) \\ (AB)^2 &= \frac{1}{2}(\mathbb{1} + 2iX + (-X)^2) = iX. \end{aligned}$$

Thus we can regard AB as the square-root of X up to a phase of i. This phase can be cancelled if we choose $\alpha = -\pi/2$ and this circuit implements a Toffoli gate with just 4 CNOTs and single-qubit gates.

One can construct multiply controlled U gates, a C^n-U gate, by a cascading circuit using n control bits, Toffoli gates and $n-1$ auxiliary bits, as in Figure 7.20.

Verify that this works! The use of the Toffoli gates performs an "AND" of all the control bits, which finally controls the U gate. Also note that all the auxiliaries can be returned to their original state of $|0\rangle$ by adding the reverse of each of the actions after obtaining C^n-U.

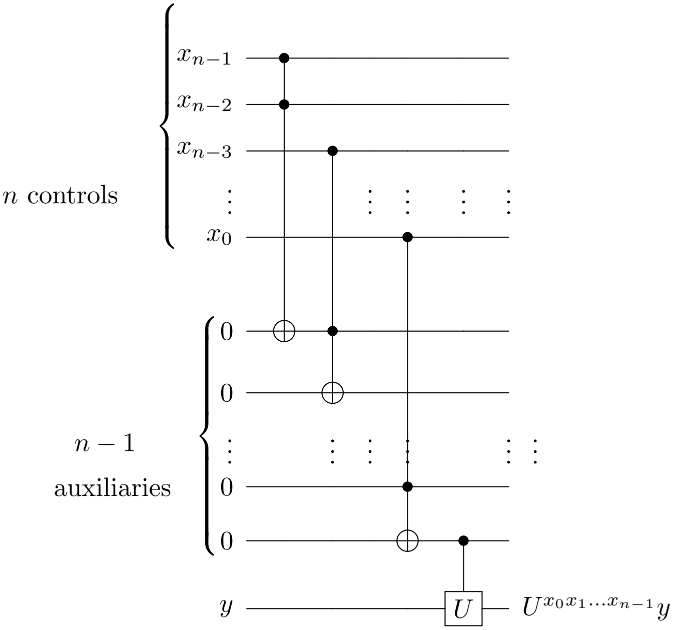


FIGURE 7.20: Implementation of C^n-U gate

7.4.2 Universal gates

We've proved that the CNOT gate along with all possible single-qubit gates form a universal set. But this set is still infinite. We'd like to do better: to get a finite set of gates as in the classical case. Of course we must realize that the set of possible single qubit gates is itself infinite as opposed to the finite number of gates in classical computation. Yet it is surprising that there exist more rigorous theorems (e.g., the Solovay–Kitaev Theorem [23]) confirming the universality of a smaller set of gates, such as for example, H, CNOT, $S = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}$ and $T = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix}$ or the Toffoli and H gates. But these sets of gates cannot be used to construct arbitrary gates to infinite precision. So the theorems actually prove that one can *approximate* arbitrary unitary gates, to *any* degree of accuracy, by using a finite set of gates. We will not dwell on these theorems or their proofs here.

7.5 Comments on Measurement

Some issues regarding measurement in quantum circuits are to be noted here, which you can prove for yourself with some thought:

1. **Deferred measurement**: when measurements are made in a circuit and after that further gates are implemented (whether controlled by the measurement or no), it can always be assumed that the measurement is made at the very end of the circuit. This is saying that measurement can always be assumed to have been *deferred* to the end of the computation without any effect on the results.

2. **Implicit measurement**: any quantum wires that are left at the end of the circuit can be assumed to have been measured: their states will anyway have collapsed when other wires are measured for the purpose of readout.

3. **Irreversibility**: quantum measurement is in general an irreversible process, and if included in a circuit, will make it irreversible. However, if the measurement reveals no information about the state being measured (refer for instance to the teleportation protocol of Example 7.2) then the circuit is still reversible!

Many of the results in this chapter are discussed in the paper by Barenco et al. [3], and in the book by Mermin [48].

Problems

7.1. Show that the n-qubit Hadamard gate acts as

$$H^{\otimes n}|x\rangle_n = \frac{1}{\sqrt{2^n}} \sum_{y=1}^{2^n-1} (-1)^{x \cdot y} |y\rangle. \tag{7.28}$$

where $x \cdot y$ is the bitwise product of x and y:

$$x \cdot y = x_0 y_0 \oplus x_1 y_1 \oplus \ldots x_{n-1} y_{n-1}. \tag{7.29}$$

7.2. Often it helps to simplify circuits when we can identify equivalences between some combinations of gates. Prove, for example, the following circuit identities:

(a) $HXH = Z$

(b) $HYH = -Y$

(c) $HZH = X$

7.3. Show the following relations concerning rotation matrices:

(a) $R_n(\theta_1)R_n(\theta_2) = R_n(\theta_1 + \theta_2)$

(b) $XR_n(\theta)X = R_n(-\theta)$

7.4. The "SWAP" gate S interchanges two inputs, defined by

$$S|xy\rangle = |yx\rangle.$$

(a) Give the matrix representing this gate.

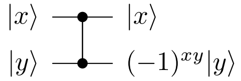
(b) Show that it can be implemented by 3 CNOT gates as

$$S_{12} = C_{12}C_{21}C_{12}.$$

(c) Show that the matrix is equivalent to

$$S_{12} = \frac{1}{2}\left(\mathbb{1} + X_1X_2 + Y_1Y_2 + Z_1Z_2\right)$$

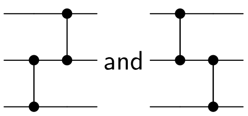
7.5. The controlled phase-flip gate takes $|11\rangle$ to $-|11\rangle$ while leaving the other basis states unchanged. It is sometimes represented as follows, since its action is symmetric in the inputs:



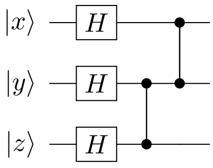
(a) Construct the matrix for this gate.

(b) Build a CNOT gate using controlled phase-flip gates an another single-qubit gate.

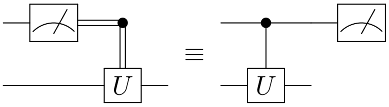
(c) What is the difference in the outputs of the following two circuits?



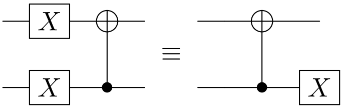
(d) Evaluate the output of the circuit

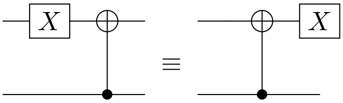


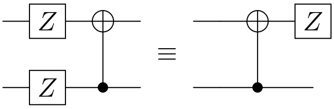
7.6. Show that classical conditional operations are equivalent to quantum control, i.e., show that the following two circuits are equivalent:

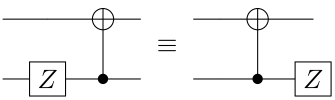


7.7. Verify the following circuit identities:

(a)


(b)


(c)


(d)


7.8. Consider the four possible 1-bit functions

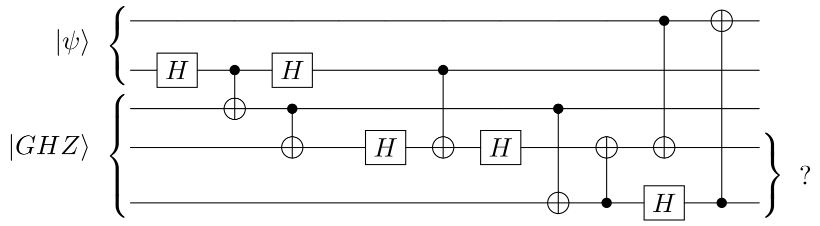
$$f_0 : \begin{array}{l} 0 \to 0 \\ 1 \to 0 \end{array}, \quad f_1 : \begin{array}{l} 0 \to 0 \\ 1 \to 1 \end{array}, \quad f_2 : \begin{array}{l} 0 \to 1 \\ 1 \to 0 \end{array}, \quad f_3 : \begin{array}{l} 0 \to 1 \\ 1 \to 1 \end{array}.$$

Construct the matrix representation of U_f for each. Also give a simple circuit to implement each using basic 1-qubit gates.

7.9. Consider 1-bit integer addition. Write down the truth tables for sum and carry bits. Then construct a quantum half-adder by implementing the truth tables, using only CNOT gates.

7.10. Examine the following circuit and analyze the final output. Here, the input is an unknown entangled state

$$\begin{aligned} |\psi\rangle &= \alpha|01\rangle + \beta|10\rangle \\ \text{and } |GHZ\rangle &= \frac{1}{\sqrt{2}}\left(|000\rangle + |111\rangle\right). \end{aligned}$$



Chapter 8

Quantum Algorithms

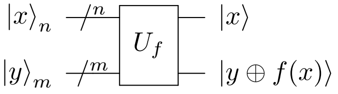
In the last chapter, we introduced the circuit model for quantum computation, where a computation is essentially the evaluation of a function. In the binary system of computation a function is a map $f : \{0,1\}^n \mapsto \{0,1\}^m$. The function takes an n-bit input and produces an m-bit output. However, this can be regarded as m functions $f_i : \{0,1\}^n \mapsto \{0,1\}$, that take an n-bit input and give a one-bit output, each of which is one bit of $f(x)$. We can thus restrict our attention to $n \to 1$ functions alone, in other words, we can reduce our problem to questions with yes/no answers, a so-called binary decision problem.

In order to evaluate a function, we have to feed it all allowed inputs and tabulate the corresponding outputs. We can construct circuits evaluating different functions by suitable combinations of gates. Now a given function evaluation or computational task is to be optimized by exploiting quantum mechanics. How is this done efficiently?

We have seen that the quantum function evaluator for $f : \{0,1\}^n \mapsto \{0,1\}^m$ is defined by the 2^{n+m}-square unitary operator of Definition 7.1:

$$U|x\rangle_n|y\rangle_m = |x\rangle_n|y \oplus f(x)\rangle_m.$$

This is essentially an **f-controlled NOT gate** as in Figure 7.15:



For $y = 0$ the output is simply $|f(x)\rangle$. If the function evaluator is fed a uniform superposition of all n-qubit basis states, then the linearity of the operator U_f ensures that the output is a uniform superposition of functions on each input, (entangled with the corresponding input state,) as in Figure 8.1.

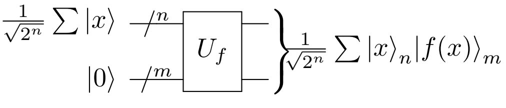

$\frac{1}{\sqrt{2^n}}\sum|x\rangle$
n
$|0\rangle$
m
U_f
$\frac{1}{\sqrt{2^n}}\sum|x\rangle_n|f(x)\rangle_m$

FIGURE 8.1: The quantum function evaluator with a uniform superposition.

Here the top line is sometimes called the *input register*, since the input to the function is fed through it, and the bottom one is the so-called *output register*, since it reflects the state containing the evaluated function.

The advantage of the quantum function evaluator is that it can take all possible inputs simultaneously as a superposition of states, and the corresponding outputs are all simultaneously present in the output state. This has often been called quantum parallelism. However, in this basic form it gives us no advantage, since to actually discover the value of the function, we must measure the output, upon which the output state will collapse to *one* of the possible outputs at random. The trick to making quantum computing work is to cleverly manipulate this basic function evaluator in such a way that the probability amplitude for the answer to the problem is maximum. It is quantum interference that enables this to happen. If this had not been possible, quantum computing would have been a forgotten chapter in the history of science. As it happens, this field received new impetus when Peter Shor shook up the world in 1994 with his famous algorithm for finding the prime factors of large integers.

All known quantum algorithms seem to fall into three broad classes:

1. Based on the Fourier transform: Deutsch–Josza, Shor's algorithm etc.

2. Based on quantum search, involving amplitude amplification: Grover's algorithm etc.

3. Quantum simulations.

In this chapter we will examine the first two kinds, leaving the last to more physics-specific texts. The algorithms are typically framed as yes-no answers to inputs to the function evaluator treated as a black box (Figure 8.2). This is also referred to as querying the oracle, as the unknown function evaluator is regarded, like a mysterious priestess who will only give single-bit answers when questioned!

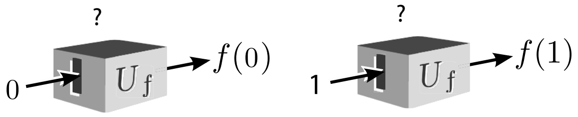


FIGURE 8.2: Classical black box function evaluator as an oracle.

8.1 The Deutsch Algorithm

Let's start with 1-bit functions $f : \{0,1\} \mapsto \{0,1\}$. There are totally four possible functions, and evaluated on inputs 0 and 1 can give answers 0 or 1. To actually determine which of these our black box is we need to query it with both inputs, whether classically or otherwise, and we obtain no advantage using quantum computing. However, as David Deutsch [24] showed in 1985, it is possible to distinguish the function on the basis of some property, more efficiently in the quantum case. The particular classification Deutsch's algorithm considers is the following: they are either *constant* ($\mathbb{C}$), i.e., $f(0) = f(1)$, or *balanced* ($\mathbb{B}$), i.e., the outputs contain an equal number of 0's and 1's ($f(0) = \overline{f(1)}$).

Example 8.1.1. For $n > 1$, functions need not fall into the classes $\mathbb{C}$ or $\mathbb{B}$ alone. For example, consider f_1 and f_2 defined by:

$$f_1 : \begin{array}{rl} f(00) = & 0 \\ f(01) = & 1 \\ f(10) = & 0 \\ f(11) = & 1 \end{array}, \quad f_2 : \begin{array}{rl} f(00) = & 0 \\ f(01) = & 1 \\ f(10) = & 0 \\ f(11) = & 0 \end{array}$$

Here, f_1 is balanced while f_2 is neither constant nor balanced.

Deutsch's algorithm[1] is formulated for the following problem. Although it might seem contrived, it is the first algorithm to demonstrate the principles of the new paradigm.

The problem: given a black-box (oracle) that implements a 1-bit function $f(x)$, how will you determine whether the function belongs to class $\mathbb{C}$ or to class $\mathbb{B}$ with a minimum number of runs of the black box (or equivalently, queries to the oracle)?

Classically, it is clear that we have to run the machine twice, with inputs 0 and 1.

The Deutsch algorithm shows how this problem can be solved in just *one* run of the black box. The circuit is shown in Figure 8.3, that we will work through step by step.

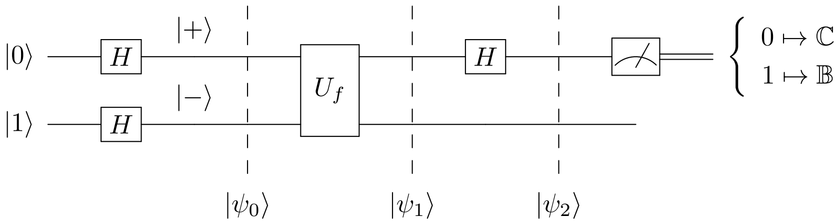


FIGURE 8.3: The Deutsch algorithm.

Step 1: Supply as input the uniform superposition

$$H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle). \tag{8.1}$$

[1] The presentation given here is not the original one in [24] but an improved version presented first by [19].

Step 2: On the bottom register, supply the state $H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$. This is the crucial feature that introduces useful interference in the result. The reason for this will be clear when we evaluate the output of the black box. So the input state is

$$\begin{aligned} |\psi_0\rangle &= \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \\ &= \frac{1}{2}[|00\rangle - |01\rangle + |10\rangle - |11\rangle] \end{aligned} \tag{8.2}$$

Step 3: Run the function evaluator. The output is

$$\begin{aligned} |\psi_1\rangle &= U_f|\psi_0\rangle \\ &= \frac{1}{2}\left[|0\rangle|f(0)\rangle - |0\rangle|\overline{f(0)}\rangle + |1\rangle|f(1)\rangle - |1\rangle|\overline{f(1)}\rangle\right] \\ &= \frac{1}{2}|0\rangle\left[|f(0)\rangle - |\overline{f(0)}\rangle\right] + \frac{1}{2}|1\rangle\left[|f(1)\rangle - |\overline{f(1)}\rangle\right]. \end{aligned} \tag{8.3}$$

Step 4: Measure the top register in the X-basis. That is, change basis by applying the H gate on the first qubit and then measure it. Just before the measurement, the output state on both wires is

$$\begin{aligned} |\psi_2\rangle &= H_1|\psi_1\rangle \\ &= \frac{1}{2\sqrt{2}}|0\rangle\left[|f(0)\rangle - |\overline{f(0)}\rangle + |f(1)\rangle - |\overline{f(1)}\rangle\right] \\ &\quad + \frac{1}{2\sqrt{2}}|1\rangle\left[|f(0)\rangle - |\overline{f(0)}\rangle - |f(1)\rangle + |\overline{f(1)}\rangle\right] \end{aligned} \tag{8.4}$$

If the function is $\mathbb{C}$, then $f(0) = f(1)$ and the amplitude for $|0\rangle$ is 1 while that for $|1\rangle$ is 0. On the other hand, when f is $\mathbb{B}$ then $f(0) = \overline{f(1)}$ and the amplitude for $|1\rangle$ is 1 while that for $|0\rangle$is 0. Thus a measurement of the output gives us the answer to the query with certainty. We have run the function evaluator only once. The quantum advantage has given us a double speedup in this case.

The reason why this works is that Step 2 implements the so called "*phase kickback*" trick. If the state $|-\rangle$ on the lower register fed into the black box, then the output acquires a phase that depends on $f(x)$. This phase can effectively be regarded as attached to the state of the upper register.

$$\begin{aligned} U_f\left[|x\rangle \otimes \frac{(|0\rangle - |1\rangle)}{\sqrt{2}}\right] &= \frac{1}{\sqrt{2}}|x\rangle\left[|f(x)\rangle - |\overline{f(x)}\rangle\right] \\ &= \begin{cases} \frac{1}{\sqrt{2}}|x\rangle(|0\rangle + |1\rangle) & \text{if } f(x) = 0, \\ \frac{1}{\sqrt{2}}|x\rangle(|1\rangle - |0\rangle) & \text{if } f(x) = 1 \end{cases} \\ &= (-1)^{f(x)}|x\rangle\frac{1}{\sqrt{2}}\left[|0\rangle - |1\rangle\right]. \end{aligned} \tag{8.5}$$

The output is separable, with the lower register unchanged in state $|-\rangle$, while the upper register is effectively the input with an $f(x)$-dependent phase.

With this effect, we can re-analyze the algorithm with the uniform superposition $|x\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ in the input register:

$$\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \xrightarrow{U_f} \frac{1}{\sqrt{2}}\left[(-1)^{f(0)}|0\rangle + (-1)^{f(1)}|1\rangle\right] \tag{8.6}$$

$$\begin{aligned} &\xrightarrow{H} \frac{1}{\sqrt{2}}\left[\left((-1)^{f(0)} + (-1)^{f(1)}\right)|0\rangle\right. \\ &\qquad \left. + \left((-1)^{f(0)} - (-1)^{f(1)}\right)|1\rangle\right] \end{aligned} \tag{8.7}$$

where it's obvious that a measurement gives $|0\rangle$ if $f(x)$ is $\mathbb{C}$ and $|1\rangle$ if $f(x)$ is $\mathbb{B}$.

8.1.1 Deutsch–Josza algorithm

The Deutsch algorithm was extended to n-bit functions by Josza and others in 1992 [26].

The problem: Given an $n \to 1$ function $f : \{0,1\}^n \mapsto \{0,1\}$ that is guaranteed to be either constant or balanced, find out which it is in a minimum number of runs.

Classically, we would proceed by querying the oracle with each n-bit number. If we find an answer that is not equal to the previous one then we have a balanced function. In worst-case scenario, we might find the same $f(x)$ until the half the possible inputs, i.e., after querying the function $2^n/2$ times. The answer to the next query would solve the problem. Thus we need to run the oracle at worst $2^{n-1} + 1$ times: exponential in the number of bits of input.

The quantum algorithm achieves the distinction in just one run! This is a dramatic speedup indeed. The circuit (Figure 8.4) is an n-qubit extension of that for the Deutsch problem:

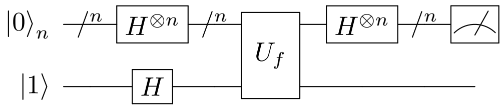


FIGURE 8.4: The circuit for the Deutsch–Josza algorithm.

The input to the circuit is the uniform n-qubit superposition

$$H^{\otimes n}|0\rangle_n = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle. \tag{8.8}$$

Due to the phase-kickback trick, the output of U_f on the input register is the superposition

$$|\psi_1\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} (-1)^{f(x)} |x\rangle. \tag{8.9}$$

After the Hadamard, this state becomes

$$H^{\otimes n}|\psi_1\rangle = \frac{1}{2^n} \sum_{x,y=0}^{2^n-1} (-1)^{[f(x)+x\cdot y]} |y\rangle. \tag{8.10}$$

Here, we have used the result of Equation 7.28 and $x \cdot y$ is the bitwise product of x and y summed modulo 2, as in Equation 7.29. Now when $f(x)$ is constant, the amplitude for the state $|0\rangle$ in this superposition is

$$\text{coefficient of } |0\rangle = \frac{1}{2^n} \sum_x (-1)^{f(x)} = 1.$$

In other words the probability of getting $|0\rangle$ is one for a constant function. Whereas if $f(x)$ is balanced, then the amplitude for $|0\rangle$ is a sum of an equal number of $+1$s and -1s, that is, zero. Thus if the function is balanced, the output measures to any number *other* than 0. We thus distinguish the two classes in one run of the black box, which is nearly an n-fold speedup compared to the classical case.

8.2 The Bernstein–Vazirani Algorithm

We'll now look at algorithms that show more substantial speedups compared to classical ones. One such algorithm was invented by Umesh Vazirani and his student Ethan Bernstein in 1993 [11]. This algorithm identifies a linear Boolean function in one query of the oracle.

The problem: given a function evaluator for

$$f : \{0,1\}^n \mapsto \{0,1\} \text{ where } f(x) = a \cdot x, \quad a \in [0, 2^n], \tag{8.11}$$

and the dot is a bitwise product with modulo 2 addition:

$$a \cdot x \equiv a_0 x_0 \oplus a_1 x_1 \oplus \cdots \oplus a_{n-1} x_{n-1}, \tag{8.12}$$

determine the function, or in other words find a.

Example 8.2.1. An example of such a function for $n = 2$ and $a = 11$, which evaluates to

$$f(00) \quad = \quad 0$$

$$
\begin{aligned}
f(01) &= 0.1 \oplus 1.1 = 1 \\
f(10) &= 1.1 \oplus 0.1 = 1 \\
f(11) &= 1.1 \oplus 1.1 = 0
\end{aligned}
$$

Classically, we can determine the k^{th} bit of a if we feed the oracle the input $x = 2^k$, that has only the k^{th} bit as 1 and all the rest as 0. This becomes obvious when you look at the binary expansion of a:

$$a = a_0 + a_1 2^1 + \cdots + a_k 2^k + \ldots \implies a_k = a \cdot 2^k. \tag{8.13}$$

This calls the function n times.

The quantum algorithm, which uses the same circuit as for the Deutsch–Josza algorithm, succeeds with *one* call!

Let's analyze the output of the circuit of Figure 8.4 for this form of the function:

$$\sum_x \sum_y \frac{1}{2^n} (-1)^{f(x)+x\cdot y} |y\rangle \otimes |-\rangle \quad = \quad \frac{1}{2^n} \sum_y \left[\sum_x (-1)^{a\cdot x + y\cdot x} \right] |y\rangle \otimes |-\rangle$$
$$. \tag{8.14}$$

The amplitude for $|y\rangle$ is $\frac{1}{2^n} \sum_x (-1)^{a\cdot x + y\cdot x} = \frac{1}{2^n} \sum_x (-1)^{(a+y)\cdot x} = 1$ if $y = a$! It's easy to see why it is zero for all other values of y. Thus with certainty, the output of the circuit gives us a.

A more explicit way of seeing why this works is by analyzing the circuit for U_f. This analysis is lucidly given in Mermin [48]. The black box for $a{\cdot}x$ flips the bit in the lower register whenever a bit of the input x and the corresponding bit of a are both 1. For instance, suppose we had $a = 11010$ with $n = 5$. Then it can easily be seen that $a \cdot x$ is implemented by the circuit of Figure 8.5.

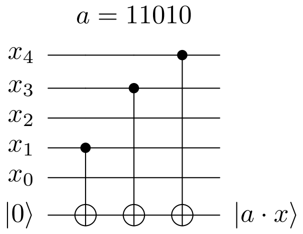


FIGURE 8.5: A circuit that executes U_f for $f = 11010 \cdot x$.

Coming to the circuit for solving the Bernstein–Vazirani problem, it has an H gate before each qubit enters the function evaluator and after. This is true even of the lower register, which can be thought of as initialized to $|1\rangle$. Note that an H gate before and after a CNOT interchanges the roles of the control and target qubits (see Figure 7.8).

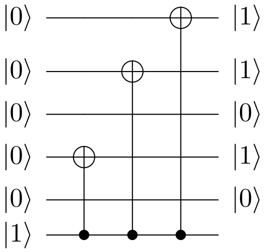


FIGURE 8.6: Analysis of circuit for the Bernstein–Vazirani algorithm for $a = 11010$.

The solution is therefore the circuit of Figure 8.6, whose output directly reads out the bits of a.

The algorithm thus gives an n-fold speedup over the classical case.

8.3 Simon's Algorithm

Even though the Bernstein–Vazirani algorithm offers such a great speedup, the classical solution is still not exponential. Daniel Simon came up with an algorithm [65] in 1994 that is the first to demonstrate a dramatic exponential speedup over a hard classical problem, but the solution is probabilistic. This feature is characteristic of many quantum algorithms. Simon's problem also illustrates a class of problems that basically use Fourier transforms, in the form of the amplitudes of the output states that "interfere" to give a large probability for the expected solution.

The problem: Given a black box implementing a function

$$f : \{0,1\}^n \mapsto \{0,1\}^{n-1} \text{ such that} f(x \oplus a) = f(x), \quad a \in [0, 2^n - 1], \quad (8.15)$$

determine a with the minimum number of queries to the box.

Example 8.3.1. The functions considered in Simon's algorithm can be thought of as "periodic" under bitwise addition. For example, let's look at the 3-bit function

x	000	001	010	011	100	101	110	111
$f(x)$	3	2	2	3	1	4	4	1

The first repetition is of the value $f(1) = f(2)$. The "period" is therefore $a = 001 \oplus 010 = 011 = 3$. You can verify that all the other repetitions also satisfy the same condition.

The classical solution to this problem is *hard*, i.e., the number of runs of the function grows exponentially as the size of the input. We would query the oracle with successive values of n-bit numbers x until we found a repeated value for the output: $f(x_i) = f(x_j)$. Then we could calculate $a = x_i \oplus x_j$. However, a could be any one of 2^n possible numbers. By the m^{th} run, $\frac{1}{2}m(m-1)$ pairs have been compared and eliminated as possible a's. For reasonable chance of success, we need $\frac{1}{2}m(m-1) \geq 2^n \implies$ a lower bound on the number of trials $m = \Omega(2^{n/2})$, which is exponential in the number of bits.

The quantum circuit (Figure 8.7) that solves this problem is essentially the same as the Deutsch–Josza circuit except that the lower register is also expanded to n qubit, and initialized to $|0\rangle_n$ (we dispense with the phase kickback).

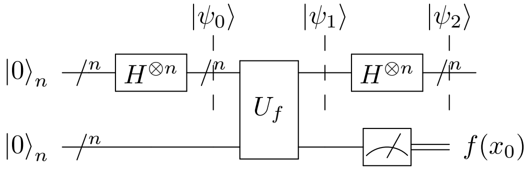


FIGURE 8.7: The circuit for the Simon algorithm.

The input to the oracle gives us

$$U_f|\psi_0\rangle \otimes |0\rangle = U_f \left[\sum_{x=0}^{2^n-1} |x\rangle \otimes |0\rangle \right] = \sum_{x=0}^{2^n-1} |x\rangle \otimes |f(x)\rangle. \tag{8.16}$$

In order to analyze the solution, let us use the reverse of the principle of delayed measurement, and assume we measure the lower register after the action of U_f. Let's denote the outcome by $f(x_0)$, which is generated from two possible inputs x_0 or $x_0 \oplus a$. The top register therefore collapses to a superposition of these two states alone:

$$|\psi_1\rangle = \frac{1}{\sqrt{2}} \left(|x_0\rangle + |x_0 \oplus a\rangle \right). \tag{8.17}$$

If we now apply H to each qubit in the upper register, we get

$$|\psi_3\rangle = \frac{1}{\sqrt{2}} \frac{1}{\sqrt{2^n}} \sum_{y=0}^{2^n-1} \left[(-1)^{x_0 \cdot y} + (-1)^{(x_0 \oplus a) \cdot y} \right] |y\rangle. \tag{8.18}$$

Now $a \cdot y$ is either 0 or 1. If $a \cdot y = 1$, then the amplitude for $|y\rangle$ is zero and all those states do not occur in the output. Thus the only states that can be measured in the output are those for which the condition $a \cdot y = 0$ is satisfied. This is a binary algebraic equation with n unknowns (the bits of a). We can find a if we can obtain n independent equations, corresponding to n different values of y. If we repeat the experiment until we have collected n distinct, non-zero y's then we can solve for the bits of a. It is not guaranteed that we will get a distinct y on each run, so we may most probably have to run the oracle more than n times.

To determine the complexity of this problem, we will need to estimate how the number of runs of the oracle scales with n. It can be shown (see Box 8.3) that the number of times the oracle has to be queried is $n + m$ where m doesn't depend on n. This algorithm is thus a sub-exponential solution to a classically hard problem.

Box 8.1: Complexity Analysis for Simon's Algorithm

As in many quantum algorithms, the analysis of why the algorithm is computationally more efficient than the classical case involves a detailed mathematical examination of the solution. In the case of Simon's algorithm, the output after measurement is an n-bit string y such that

$$a \cdot y = a_{n-1}y_{n-1} \oplus a_{n-2}y_{n-2} \oplus \cdots \oplus a_1y_1 \oplus a_0y_0 = 0.$$

We need to collect at least $n-1$ such *distinct* bit-strings in order to determine the coefficients a. So we need to query the oracle at least $n-1$ times and need to find a lower bound on the probability of success.

Suppose we ran the algorithm k times and got linearly independent y's. What's the probability that the next run gives a different y? The minimum probability for this occurring is

$$\frac{2^n - 2^k}{2^n} = 1 - 2^{k-n}.$$

So the probability of getting $n-1$ independent y's is just

$$\mathcal{P} = \left(1 - \frac{1}{2^n}\right)\left(1 - \frac{1}{2^{n-1}}\right)\cdots\left(1 - \frac{1}{2}\right).$$

Now notice that $(1-s)(1-t) = 1 - (s+t) + st \geq 1 - (s+t)$. So we have

$$\begin{aligned} \mathcal{P} &\geq \left(1 - \sum_{i=1}^{n} \frac{1}{2^i}\right)\frac{1}{2} \\ &\geq \left(1 - \frac{1}{2}\right)\frac{1}{2} = \frac{1}{4}. \end{aligned}$$

This means that there is a *finite* minimum probability with which we will succeed in n runs. To ensure success, we'll have to run the algorithm a few more times, *independent* of n, so that the number of runs is still $\mathcal{O}(n)$.

The trick that makes the kind of algorithms considered so far work is that the output before measuring in the X basis has $f(x)$-dependent phases. Until now these phases were restricted to ±1. More general phases come about if the Fourier transform is implemented. In this section, we introduced the idea of using the H gate on the output to produce interfering amplitudes. This is just a special case of the quantum Fourier transform, as we will see in the next section.

8.4 Quantum Fourier Transform and Applications

The Fourier transform, a mathematical tool named after the 18th century French mathematician Joseph Fourier, is an invaluable tool in engineering and the sciences. No technical education is complete without a firm grasp of this technique and its uses. The simplest way to understand the Fourier transform $\mathcal{F}$ of a function $f(x)$ is to imagine the function as made up of various components that are periodic (like a sine function) with a frequency y, and $\mathcal{F}(f(x))$ as a function $\tilde{f}(y)$ measuring the amplitude of each frequency component in the function. In other words, we construct a decomposition of the function in terms of the oscillatory exponential $e^{-2\pi iyx}$, where the coefficients in that decomposition are the Fourier transform:

$$f(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} dy\; e^{2\pi iyx} \tilde{f}(y). \tag{8.19}$$

This formula is said to define the *inverse Fourier transform* of $\tilde{f}(y)$, while the Fourier transform is defined as

$$\mathcal{F}(f(x)) = \tilde{f}(y) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} dx\; e^{-2\pi iyx} f(x). \tag{8.20}$$

The factor in front of the integral captures the normalization. A function can in general have an infinite number of frequency components, and the frequencies can be distributed continuously. That's how the Fourier transform is a continuous function of the frequency y.

The two Equations 8.20 and 8.19 define a *Fourier transform pair.* The Fourier transform naturally produces complex numbers, so that $f(x)$ and $\tilde{f}(y)$ are in general complex. When we compute the Fourier transform on a digital machine, we need to discretize the integral to get the Discrete Fourier Transform (DFT).

8.4.1 The discrete Fourier transform and classical algorithm

When $f(x)$ is a discrete function over the finite range $N = 2^n$ of discrete n-bit inputs x, we can think of it as a vector with N components $\{f_0\; f_1\; ...\; f_{N-1}\}$. The integral over x in Equation 8.20 is then a sum over an index k with

$x \to k/N$ and the limits are restricted from 0 to $N-1$. We then get the discrete Fourier transform of order N defined as

$$\tilde{f}(y) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{-2\pi i y k/N} f_k.$$

This is another vector with N components $\{g_0\ g_1\ ...\ g_{N-1}\}$, given by

$$g_j = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{-2\pi i j k/N} f_k. \tag{8.21}$$

These are just the coefficients of orthogonal harmonic components $e^{2\pi i j k/N}$ of the function, which can be expressed as the inverse discrete Fourier transform (IDFT):

$$f_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} e^{2\pi i j k/N} g_j. \tag{8.22}$$

We can regard the DFT as a complex matrix transformation of the vector $\{f_k\}$:

$$g_j = \sum_{k=0}^{N-1} M_{jk} f_k; \qquad f_k = \sum_{j=0}^{N-1} M^{-1}_{jk} g_k, \tag{8.23}$$

where M_{jk} are the elements of an $N \times N$ matrix M given by

$$M_{jk} = \frac{1}{\sqrt{N}} e^{2\pi i j k/N} = \frac{1}{\sqrt{N}} \omega_N^{jk}. \tag{8.24}$$

Here $\omega_N = e^{2\pi i/N}$ is the N^{th} root of unity. More explicitly,

$$\begin{pmatrix} g_0 \\ g_1 \\ \vdots \\ g_{N-1} \end{pmatrix} = \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega_N & \omega_N^2 & \cdots & \omega_N^{N-1} \\ 1 & \omega_N^2 & \omega_N^4 & \cdots & \omega_N^{2(N-1)} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \omega_N^{N-1} & \omega_N^{2(N-1)} & \cdots & \omega_N^{(N-1)^2} \end{bmatrix} \begin{pmatrix} f_0 \\ f_1 \\ \vdots \\ f_{N-1} \end{pmatrix} \tag{8.25}$$

Example 8.4.1. The simple case of $N = 2$, $\omega_2 = e^{i\pi} = -1$ and

$$\mathrm{DFT}_2 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & \omega_2 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{8.26}$$

which is just the Walsh–Hadamard transform.

Exercise 8.1. Write out the DFT matrix for $N = 4$.

Exercise 8.2. Calculate the DFT on the N-dimensional zero-vector.

Example 8.4.2. Unitarity of the DFT:

The crucial point that allows us to extend the DFT to an operator on quantum states is that it is unitary. To prove this, we need to show that

$$M^\dagger M = \mathbb{1} \implies \sum_{l=0}^{N-1} M_{jl} M^*_{lk} \quad = \quad \delta_{jk} \tag{8.27}$$

where M_{jk} is defined by Equation 8.24.

$$\text{When } j = k: \quad \frac{1}{N} \sum_l \omega_N^{jl} \omega_N^{-lj} = \frac{1}{N} \sum_l 1 = 1; \tag{8.28}$$

When $j \neq l$, then $\sum_l \omega_N^{l(j-k)}$ is the sum of N terms of a geometric series whose first term is 1 and ratio is $\omega_N^{(j-k)}$. So we have

$$\sum_l \omega_N^{l(j-k)} = \frac{1 - \omega_N^{N(j-k)}}{1 - \omega_N^{(j-k)}} = 0 \tag{8.29}$$

Thus Equation 8.27 is proved.

Box 8.2: Classical FFT Algorithm

Computing the DFT_N of a vector involves evaluating N^2 elements of the DFT matrix, and looks like a job that scales as 2^{2n} with the number of bits $n = \log_2 N$. In implementing the DFT transform on a digital machine, one can easily optimize by exploiting the properties of the integer powers of ω_N. There are cycles among elements of DFT_N, since $\omega_N^N = 1$. So while a direct matrix multiplication of the form of Equation 8.24 would typically require $\mathcal{O}(N^2)$ basic operations, the optimized fast Fourier transform (FFT) algorithm requires $\mathcal{O}(N \log_2 N)$ operations only.

For example, consider $N = 4$; $\omega_4 = e^{2\pi i/4} = i$, $\omega_4^2 = -1$, $\omega_4^4 = 1$.

$$\mathrm{DFT}_4 = \frac{1}{2} \left(\begin{array}{cc:cc} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ \hdashline 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{array} \right) \tag{8.30}$$

Now there is a relationship between the upper and lower halves of this matrix. Look at the highlighted columns, repeated for the upper and lower halves: they form a 2 × 2 matrix that acts on the even index components (note the index starts at 0).

$$\frac{1}{2}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = \frac{1}{\sqrt{2}}\mathrm{DFT}_2. \tag{8.31}$$

The part that acts on the odd index components is for the upper half

$$\begin{aligned} \frac{1}{2}\begin{pmatrix} 1 & 1 \\ i & -i \end{pmatrix} &= \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \times \mathrm{DFT}_2 \\ &= \frac{1}{\sqrt{2}}\mathrm{Diag}(1\ \omega_4) \times \mathrm{DFT}_2. \end{aligned} \tag{8.32}$$

The negative of this acts on the lower half. Thus the DFT of a 4-d vector is reduced to two DFT's of a 2-d vector. This is at the heart of the classical FFT algorithm.

The above example shows that DFT_N can be reduced to $\mathrm{DFT}_{N/2}$. The FFT algorithm works by recursively dividing the original vector into even numbered and odd numbered elements, until at the final stage there are just two terms and DFT_2 can be applied. The process is then reversed by successively doubling the vectors and eventually covering the entire input. Let's see how this is possible in general: let $N = 2M$.

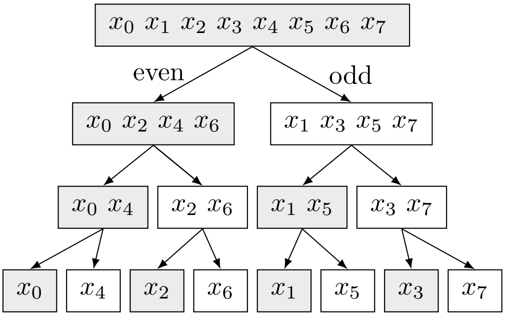
$$\mathrm{DFT}_N(f(x)) = \tilde{f}(y) = \frac{1}{\sqrt{2M}}\sum_{x=0}^{2M-1} \omega_{2M}^{xy} f(x). \tag{8.33}$$

Breaking this up into even and odd terms,

$$\begin{aligned} \mathrm{DFT}_{2M}(f(x)) =& \frac{1}{\sqrt{2M}}\left(\sum_{x=0}^{M-1} \omega_{2M}^{2xy} f(2x) + \sum_{x=0}^{M-1} \omega_{2M}^{(2x+1)y} f(2x+1)\right) \\ =& \frac{1}{\sqrt{2}}\Bigg(\underbrace{\frac{1}{\sqrt{M}}\sum_{x=0}^{M-1} \omega_M^{xy} f(2x)}_{\mathrm{DFT}_M \text{ of even terms}} + \underbrace{\frac{1}{\sqrt{M}}\sum_{x=0}^{M-1} \omega_M^{(x)y} f(2x+1)}_{\mathrm{DFT}_M \text{ of odd terms}} \times \omega_{2M}^{y}\Bigg) \end{aligned} \tag{8.34}$$

At any stage l of evaluating the DFT, one can divide the input into two to write it in terms of $\mathrm{DFT}_{l/2}$, and continue successively until one is left with DFT_2's.

Successive division of the terms in the input into two until we reach the two-term pairs is called *decimation*. The process of decimating higher-order DFT's looks like the following for $N = 8$:



We then start evaluating upward from the 2-point DFTs, successively doubling at each stage. The generic 2-point DFT looks like Figure 8.8, called a butterfly diagram for its symmetric structure. The labels on the sides represent the multiplicative factors and two lines joined at a node represent addition of the corresponding terms.

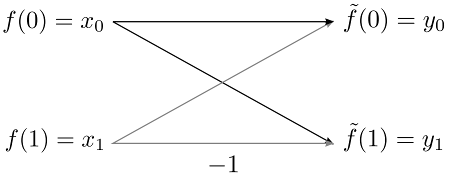


FIGURE 8.8: The 2-point DFT: butterfly diagram.

For $N = 8$, we have worked out the decimation process in Example 8.4.4. The butterfly diagram looks like Figure 8.9.

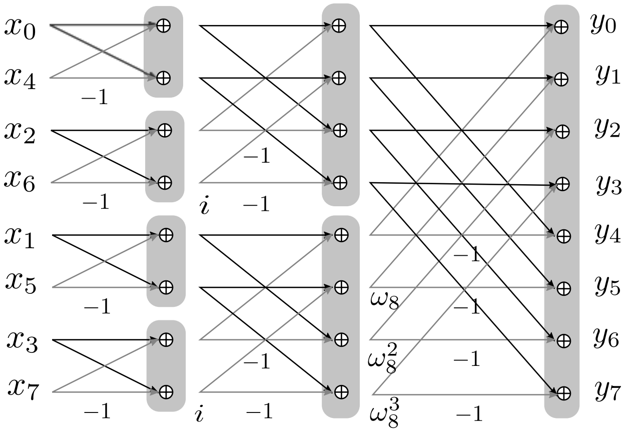


FIGURE 8.9: Butterfly diagram for computing an 8-point DFT.
The output vector $\{y_0\ y_1 \dots\ y_8\}$ is the DFT of the input vector.

Exercise 8.3. Show how the DFT of a 6-d vector reduces to the DFT's of the even and odd indexed 3-d components.

8.4.2 Complexity of the classical FFT algorithm

Suppose the computation of DFT_N requires $T(N)$ basic operations. From Equation 8.34, we see that this is related to $T(N/2)$ since we need to evaluate two DFT's of order $N/2$ and also do N multiplications of the exponential factors. Thus,

$$T(N) = 2T(N/2) + \mathcal{O}(N). \tag{8.35}$$

Using this recursively to solve for $T(N)$ one gets

$$T(N) = \mathcal{O}(N \log_2 N). \tag{8.36}$$

8.5 Definition of the QFT from Discrete Fourier Transform

The quantum Fourier transform (QFT) is simply the DFT operation on the amplitudes of a quantum state. The DFT matrix is unitary, and can therefore represent a quantum transformation. We can define the QFT (order $N = 2^n$) of an n-qubit basis state $|x\rangle$ by

$$\hat{\mathcal{F}}_N|x\rangle = \frac{1}{\sqrt{N}} \sum_{y=0}^{N-1} \omega_N^{xy} |y\rangle. \tag{8.37}$$

Interestingly, as we have seen in Equation 8.26, the QFT transform for $n = 2$ is just the Hadamard gate.

When applied to a superposition state $|\psi\rangle = \sum_i C_i|i\rangle$, the QFT performs a DFT on the coefficients C_i:

$$\begin{aligned} \hat{\mathcal{F}}_N|\psi\rangle &= \sum_i C_i U_N|i\rangle \\ &= \frac{1}{\sqrt{2^n}} \sum_i C_i \sum_j \omega_N^{ij}|j\rangle && (8.38) \\ &= \sum_j \left(\mathrm{DFT}_N C\right)_j |j\rangle. && (8.39) \end{aligned}$$

where C denotes the vector of coefficients in the state representation.

An efficient algorithm for evaluating the QFT is inspired by the FFT algorithm, where we compute the bit-wise breakup of the action of $\hat{\mathcal{F}}$ on its input. Remember, $|y\rangle_n = \bigotimes_{j=0}^{n-1} |y_j\rangle$. Each y_j takes a value 0 or 1. The QFT has superpositions of $|y\rangle$ with a phase $\omega_N^{xy/N}$, containing the integer product of x and y. We want to break this up into the constituent bits, the y_js. So we write

$$xy = \left(x_0 + 2x_1 + \cdots + 2^{n-1}x_{n-1}\right)\left(y_0 + 2y_1 + \cdots 2^{n-1}y_{n-1}\right). \tag{8.40}$$

Now any product in this expansion that has a coefficient of 2^n or higher can be dropped since it would contribute unity to the phase: $\omega_N^{2n} = 1$. So we find

$$\begin{aligned} \frac{xy}{N} &= \left(\frac{x_0}{2^n} + \frac{x_1}{2^{n-1}} + \cdots + \frac{x_{n-1}}{2}\right) y_0 \\ &+ \left(\frac{x_0}{2^{n-1}} + \frac{x_1}{2^{n-2}} + \cdots + \frac{x_{n-2}}{2}\right) y_1 \\ &\vdots \\ &+ \left(\frac{x_0}{2^2} + \frac{x_1}{2}\right) y_{n-2} \\ &+ \left(\frac{x_0}{2}\right) y_{n-1}. \end{aligned} \tag{8.41}$$

Using the binary “point” notation

$$0.x_1x_2x_3...x_n = \frac{x_1}{2} + \frac{x_2}{2^2} + \frac{x_3}{2^3} + ...\frac{x_n}{2^n}, \tag{8.42}$$

$$\frac{xy}{N} = y_0(0.x_{n-1}x_{n-2}\cdots x_0) + y_1(0.x_{n-2}\cdots x_0) + \cdots + y_{n-1}(0.x_0). \tag{8.43}$$

Using this to write the QFT in bit-wise expansion, we can associate an exponential factor with each bit of y, the output becomes the following product state:

$$\begin{aligned} \hat{\mathcal{F}}_N|x\rangle &= \frac{1}{\sqrt{2^n}} \sum_y e^{2\pi i xy/N} |y_0\rangle \otimes |y_1\rangle \cdots |y_{n-1}\rangle \\ &= \frac{1}{\sqrt{2^n}} \left(\sum_{y_0=0,1} e^{2\pi i y_0(0.x_{n-1}x_{n-2}\cdots x_0)} |y_0\rangle \right) \\ &\quad \otimes \left(\sum_{y_1=0,1} e^{2\pi i y_1(0.x_{n-2}\cdots x_0)} |y_1\rangle \right) \otimes \\ &\quad \cdots \otimes \left(\sum_{y_{n-1}=0,1} e^{2\pi i y_{n-1}(0.x_0)} |y_{n-1}\rangle \right) \\ &= \left(\frac{|0\rangle + e^{2\pi i(0.x_{n-1}x_{n-2}\cdots x_0)}|1\rangle}{\sqrt{2}} \right) \otimes \left(\frac{|0\rangle + e^{2\pi i(0.x_{n-2}\cdots x_0)}|1\rangle}{\sqrt{2}} \right) \otimes \\ &\quad \cdots \otimes \left(\frac{|0\rangle + e^{2\pi i(0.x_0)}|1\rangle}{\sqrt{2}} \right) \end{aligned} \tag{8.44}$$

Each term in the product of Equation 8.44 is the state of an output qubit for the corresponding qubit of the input. This translates into a circuit for evaluating $\hat{\mathcal{F}}_{2^n}$. The order of occurrence of the terms must be noted: the first term is the least significant bit, and the last is the most significant bit of the output.

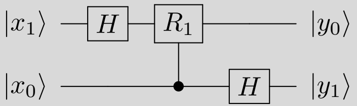
Example 8.5.1. QFT circuit for $n = 2$:

$$|x_1x_0\rangle \xrightarrow{\hat{\mathcal{F}}} \frac{|0\rangle + e^{2\pi i(0.x_0)}|1\rangle}{\sqrt{2}} \otimes \frac{|0\rangle + e^{2\pi i(0.x_1x_0)}|1\rangle}{\sqrt{2}}$$

This means

$$\begin{aligned} |x_1\rangle &\longrightarrow \frac{|0\rangle + e^{2\pi i(\frac{x_0}{2})}|1\rangle}{\sqrt{2}} = |y_1\rangle \\ |x_0\rangle &\longrightarrow \frac{|0\rangle + e^{2\pi i(\frac{x_1}{2}+\frac{x_0}{4})}|1\rangle}{\sqrt{2}} = |y_0\rangle \end{aligned}$$

Here $|y_1\rangle$ has an x_0-dependent phase of $e^{i\pi}$ for $|1\rangle$, which can be obtained by an H acting on $|x_0\rangle$. Similarly $|y_0\rangle$ has a x_1-dependent phase of $e^{i\pi}$ and an x_0-dependent phase of $e^{i\pi/2}$ for $|1\rangle$. The first is obtained by an H on $|x_1\rangle$ while the second is the x_0-controlled action of the gate $R_1 = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/2} \end{pmatrix}$:



Note that the output is to be read in reverse order!

Exercise 8.4. Work out the circuit for the QFT for $n = 3$.

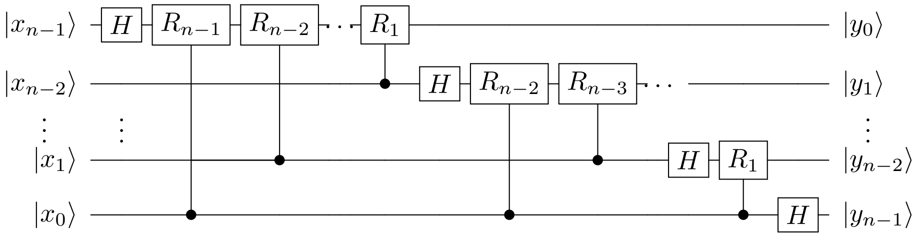


FIGURE 8.10: Circuit for the quantum Fourier transform $\hat{\mathcal{F}}_{2^n}$, on n qubits.

You should now be able to work out that Figure 8.10 is an efficient quantum circuit for the QFT on n qubits.. We require controlled phase gates, with phase matrices like

$$R_d = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/2^d} \end{pmatrix}, \tag{8.45}$$

where d can be interpreted as the distance from the control bit. Notice that the output bits are in reverse order. One can either agree to read the output in reverse order or to perform a swap at the end.

The efficiency of this circuit is related to the number of basic gate operations required per input bit. We can easily see that this is n H-gates and $n(n-1)/2$ C-R-gates for n bits, which is $\mathcal{O}(n^2)$. This is *exponentially* faster than the classical FFT which takes $\mathcal{O}(n2^n)$. Hurray for quantum algorithms!

But before we exult too much, observe that the output of the quantum Fourier transform is a superposition of basis states whose phases represent the Fourier transform of the corresponding input bit. A measurement at the end of the above circuit gives us **no** information whatsoever about the Fourier transform of the input! So we cannot use this circuit as a super-efficient Fourier transform computer! Instead, we have to incorporate it in procedures that require FT-dependent phases. And Peter Shor did just that in his path-breaking algorithm for prime factorization.

8.5.1 Period-finding using QFT

Preliminary to the Shor algorithm, let's focus on one that lends itself naturally to the QFT: computing the period r of a periodic n-bit function

$$f : \{0,1\}^n \mapsto \{0,1\}^n \text{ such that } f(x+r) = f(x), \quad r \in [0, 2^n - 1]. \tag{8.46}$$

We will take $2^n = N$ in what follows. The function could repeat more than once in the interval $[0, N-1]$, so we have

$$f(x+kr) = f(x), \qquad kr < N.$$

We assume we are presented with a black box (oracle) that evaluates such a function. The algorithm uses a circuit that is a direct extension of Simon's algorithm (Section 8.3), in which we'll use the full QFT instead of the 1-bit version (the Hadamard transform) used there. The circuit (Figure 8.11) is straightforward.

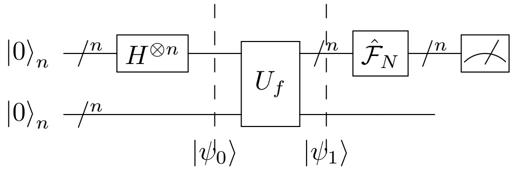


FIGURE 8.11: Circuit for quantum period finding.

The input to the U_f black box is once again

$$|\psi_0\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle_n \otimes |0\rangle_n, \tag{8.47}$$

So the output ought to be

$$|\psi_1\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle \otimes |f(x)\rangle. \tag{8.48}$$

We will again assume that we measure the lower register at this point, obtaining some number f_0. Then the top register collapses to a superposition of only those states $|x\rangle$ for which $f(x) = f_0$. All such x's are of the form $x_0 + kr$ for some $x_0 < r$, and some integer $k : kr < N$. Suppose the number of periods within the interval $[0, N-1]$ is p:

$$p = [N/r], \tag{8.49}$$

where the square bracket notation stands for the ceiling function (greatest integer less than the argument). The state of the computer is then a superposition of p terms of the form

$$|\psi_1\rangle = \frac{1}{\sqrt{p}} \sum_{k=0}^{p-1} |x_0 + kr\rangle \otimes |f_0\rangle. \tag{8.50}$$

Now subjecting the top register to a QFT, we get

$$\hat{\mathcal{F}}_N \left(\frac{1}{\sqrt{p}} \sum_{k=0}^{p-1} |x_0 + kr\rangle \right) = \sum_{y=0}^{N-1} \left(\frac{1}{\sqrt{Np}} \sum_{k=0}^{p-1} e^{2\pi i (x_0+kr)y/N} \right) |y\rangle. \tag{8.51}$$

This is a superposition of basis states with a probability of occurrence of a particular y given by the mod-squared of the term in the brackets:

$$\begin{aligned} \mathcal{P}(y) &= \frac{1}{Np} \left| \sum_{k=0}^{p-1} e^{2\pi i (x_0+kr)/N} \right|^2 \\ &= \frac{1}{Np} \left| \sum_{k=0}^{p-1} e^{2\pi i k r y/N} \right|^2 \end{aligned} \tag{8.52}$$

So y has an r-dependent probability of occurrence. The crux of this algorithm is that the most probable value of y gives us enough information about r for us to compute it. In fact, the claim is that the values of y that are measured are close to an integer multiple of N/r.

Let's first see this in the special case when there are exactly integer number of periods in the interval $[0, N-1]$, i.e., when

$$p = \frac{N}{r}.$$

We will compare the probability of y for mp when m is some integer, and when not:

$$\begin{aligned}
\mathcal{P}(y) &= \frac{1}{r}\left|\frac{1}{p}\sum_{k=0}^{p-1} e^{2\pi iky/p}\right|^2 . && (8.53)\\
\mathcal{P}(y=mp) &= \frac{1}{r}, && (8.54)\\
\mathcal{P}(y\neq mp) &= \frac{1}{rp^2}\left|\sum_{k=0}^{p-1} e^{ik\theta}\right|^2 , \text{ where } \theta = 2\pi\frac{y}{p}, && (8.55)\\
&= \frac{1}{rp^2}\frac{\sin^2(p\theta/2)}{\sin^2(\theta/2)}\\
&= 0 \ \big(\text{since } p\theta \text{ is an integer multiple of } 2\pi\big). && (8.56)
\end{aligned}$$

So the *only* values of y obtained in this case are integer multiples of N/r.

For a general function, it is highly unlikely that there are exactly integer numbers of periods in the interval $[0, N-1]$. Yet, the most probable values of y turn out to be close to integer multiples of N/r! To see this, let us start by writing

$$y = m\frac{N}{r} + \delta_m, \tag{8.57}$$

where m is an integer and $|\delta_m| \leq \frac{1}{2}$. Let's substitute this in Equation 8.52:

$$\begin{aligned}
\mathcal{P}(y) &= \frac{1}{Np}\left|\sum_{k=0}^{p-1} e^{2\pi ikr(mN/r+\delta_m)/N}\right|^2\\
&= \frac{1}{Np}\left|\sum_{k=0}^{p-1} e^{2\pi ikr\delta_m/N}\right|^2\\
&= \frac{1}{Np}\frac{\sin^2(p\theta_m)}{\sin^2\theta_m}, \text{ where } \theta_m = \frac{\pi r}{N}\delta_m. && (8.58)
\end{aligned}$$

Now since p is nearly N/r, the numerator is nearly $\sin^2(\pi\delta_m)$. Also, $r\delta_m/N$ is very small, so the denominator is nearly $\theta_m \sim \pi\delta_m r/N$.

Therefore,

$$\mathcal{P}(y) \sim \frac{1}{Np}\frac{\sin^2(\pi\delta_m)}{(\pi\delta_m r/N)^2} = \frac{1}{r}\frac{\sin^2(\pi\delta_m)}{(\pi\delta_m)^2}. \tag{8.59}$$

Since $\delta_m < 1/2$, and $\dfrac{\sin\theta}{\theta} \geq \dfrac{2}{\pi}$ for $0 \leq \theta \leq \pi/2$, (see Figure 8.12), we have

$$\mathcal{P}(y \sim m/r) \geq \frac{4}{\pi^2 r}. \tag{8.60}$$

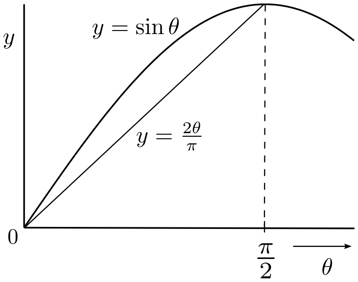


FIGURE 8.12: Graph comparing $\sin\theta$ and $2\theta/\pi$.

There are r possible such y's, so the probability of any such y is greater than $4/\pi^2 \sim 40\%$. This result is to be interpreted as saying that when we rerun the algorithm many times, with high probability we measure y's that are integer multiples of N/r. Now from such numbers we can use classical algorithms to deduce r, most famously the Euclid algorithm for continued fractions. The period-finding algorithm thus succeeds to a high probability.

Such analyses of the probability of obtaining good results are a common feature of most known quantum algorithms.

Box 8.3: Finding r Given N/r: Continued Fractions

The output y of a run of the period-finding algorithm is close to an integer multiple of N/r. Consider the number $x = y/N \sim m/r$. We now look at the continued fraction expansion of x:

$$x = c_0 + \frac{1}{x_1} = c_0 + \cfrac{1}{c_1 + \cfrac{1}{x_2}} = c_0 + \cfrac{1}{c_1 + \cfrac{1}{c_2 + \cfrac{1}{x_3}}} = \cdots \tag{8.61}$$

$$= c_0 + \cfrac{1}{c_1 + \cfrac{1}{c_2 + \cfrac{1}{c_3 + \cdots}}} \tag{8.62}$$

At each stage of the expansion (Equations 8.61), c_i is the integer part of the denominator x_i from the previous stage, and each x_i, known as the i^{th} *partial sum*, is a fraction $\in [0, 1]$. To find the fractional expression for $\frac{1}{x_i}$, Euclid's GCD algorithm can be used. Equation 8.62 is the continued fraction expansion of x. If x is a rational number then the continued fraction expansion terminates after a finite number of steps. For n-bit m and r, it turns out that the continued fraction can be computed in $\mathcal{O}(n^3)$ steps.

Now there is a theorem (proved in [50], Appendix 4) stating that m/r is one of the partial sums x_i of the continued fraction of x. $r < N$, and the best guess for r is the partial sum having the largest denominator less than N. This is tested out and if it is not the period then the we try again with a different x.

8.5.1.1 Shor's factorization algorithm

The above algorithm for period finding, due in some form to Peter Shor, is really the heart of the factorization algorithm. For the more curious, the relationship between factoring and period-finding is through a series of mathematical results that we will outline here. (This section is purely for the purpose of completeness, and the results of pure mathematics used will not be derived or explained.)

For a good understanding of what follows, one must be familiar with *modular algebra*, that is algebra restricted to the range $[0, N-1]$ by considering all results of algebraic operations as periodic with period N. Then "mod N" essentially means "the remainder after dividing the result by N". For example, addition mod 4 will mean $2+2=0$ and $2+3=1$.

- If a is a random integer $< N$ such that a and N are coprime, then it is possible to find an integer $r \in [1, N]$ such that

  $$a^r \bmod N = 1.$$

  r is called the *order* of a in mod N.

- For a with order r mod N, the function

  $$f(x) = a^x \bmod N,$$

  is periodic with period r. To see how:

  $$\begin{aligned} f(x+r) &= a^{x+r} \bmod N = (a^x \bmod N)(a^r \bmod N) \\ &= a^x \bmod N \times 1 = f(x). \end{aligned}$$

  Therefore, finding the period of a function $f(x)$ is the same as finding the order of some integer coprime with N.

- Now if N is a large integer, choose a random integer a coprime with N and find its order r using the period-finding algorithm. Now if r is even then construct $b = a^{r/2}$.

  $$\begin{aligned} b^2 &= 1 \bmod N \\ \Longrightarrow b^2 - 1 &= 0 \bmod N \end{aligned}$$

  So $b \pm 1$ must have factors common with N. If we find the GCDs of $b \pm 1$ and N we have the prime factors of N!

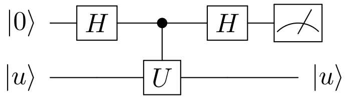
8.5.2 Phase estimation

One version of Shor's algorithm is based on phase estimation. This application of the quantum Fourier transform is used to estimate the eigenvalue of a unitary operator, which is a phase:

$$\hat{U}|u\rangle = e^{i\theta}|u\rangle; \quad \theta = 2\pi\phi \tag{8.63}$$

where ϕ is a fraction.

As a preliminary to this algorithm, let's look at a toy version. Suppose you are given U and an eigenstate $|u\rangle$. We have seen that the circuit of Figure 7.14 simulates a measurement of U.



Here,

$$\frac{1}{\sqrt{2}}\big(|0\rangle + |1\rangle\big) \otimes |u\rangle \to \frac{1}{\sqrt{2}}\big(|0\rangle + e^{2\pi i\phi}|1\rangle\big) \otimes |u\rangle. \tag{8.64}$$

If ϕ were a single bit, then you can see that the output is 0 if $\phi = 0.0$ and 1 if $\phi = 0.1$. This circuit thus gives us the value in one run. But in general ϕ will be several bits long. A measurement of the upper register in the H basis will yield a 0 or 1 with probabilities $\cos^2 \pi\phi$ and $\sin^2 \pi\phi$. A statistically large number of measurements will allow us to recover ϕ from the counts. But this is an inefficient method.

Note that the H transform on the upper register is the one-bit Fourier transform. In order to estimate ϕ to more bits of accuracy, we must have a qubit for each significant figure of ϕ and then perform an inverse Fourier transform, as shown in the circuit of Figure 8.13.

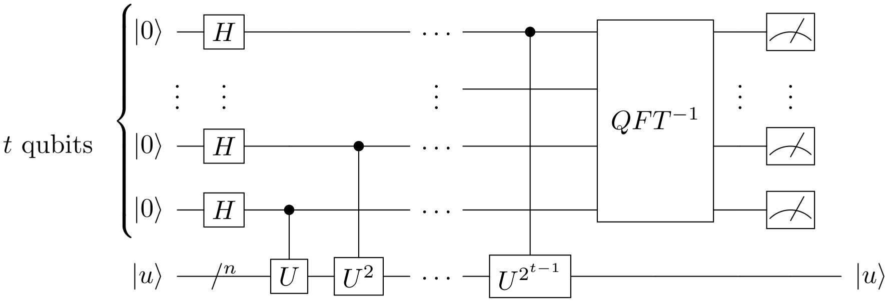


FIGURE 8.13: Circuit for phase estimation.

Imagine ϕ upto t bits as

$$\phi = 0.\phi_1\phi_2\cdots\phi_t = \frac{\phi'}{2^t}, \quad \phi' = \phi_t\phi_{t-1}\cdots\phi_1. \tag{8.65}$$

Then we start with t working qubits in the input register, and use them to control gates of the form U^{2^k}. After the control gates, the output on the k^{th} line is

$$\begin{aligned} \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \quad &\longrightarrow \quad \frac{1}{\sqrt{2}}\left(|0\rangle + e^{2\pi i 2^k \phi}|1\rangle\right) && (8.66) \\ &= \quad \frac{1}{\sqrt{2}}\left(|0\rangle + e^{2\pi i (0.\phi_1\phi_2\cdots\phi_k)}|1\rangle\right). && (8.67) \end{aligned}$$

You can see that just before the QFT gate, the state of the upper register is

$$\frac{1}{\sqrt{2^t}}\left(|0\rangle + e^{2\pi i\phi}|1\rangle\right) \otimes \left(|0\rangle + e^{4\pi i\phi}|1\rangle\right) \otimes \ldots \otimes \left(|0\rangle + e^{2\pi i.2^t\phi}|1\rangle\right) \qquad (8.68)$$

$$= \frac{1}{\sqrt{2^t}} \sum_{k=0}^{2^t-1} e^{2\pi i\phi k}|k\rangle \qquad (8.69)$$

This is just the QFT mod 2^t of ϕ' and an inverse Fourier transform will give you ϕ' exact to t significant figures.

8.6 Grover's Search Algorithm

Another famous algorithm that made a splash in the world of quantum computing was invented by L. K. Grover in 1997 [40]. This algorithm is in a different class from the ones we have studied so far, which may all be said to be QFT-based. Grover's algorithm introduced a new technique: amplitude amplification. Even though it did not demonstrate an exponential speed-up over the classical case, it was still dramatic enough to get noticed.

The problem Grover attacked was that of search for an element in an unstructured database. The problem is like doing a reverse search in a phone directory: you have a number and need to know the person it belongs to. Thus there is no regular short-cut to the search, you have to go through each entry in the book and check if it matches the number you have.

The problem can be phrased in the language of oracles, if we assume that the criterion for the search is built into a function evaluator: a function that tells you whether the input number matches the search criterion or not. So we imagine that the numbers x are indices to the entries in the database, and one index, let's say k, belongs to the entry being searched for. Then

$$f_k(x) = \begin{cases} 1 & \text{if } x = k \\ 0 & \text{otherwise.} \end{cases} \qquad (8.70)$$

Here, if x is an n-bit number, then the size of the database is $2^n = N$. As this becomes really large, the problem becomes harder. In fact, classically this problem has a complexity $\mathcal{O}(N)$. Grover's algorithm turns out to be $\mathcal{O}(\sqrt{N})$.

As in most quantum algorithms, the first step exploits quantum parallelism, and inputs the uniform superposition of all x's to the oracle U_{f_k}, the unitary implementation of $f_k(x)$:

$$|\psi\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{N-1} |x\rangle. \tag{8.71}$$

It also uses the phase kickback trick to give f-dependent phases to the states in this superposition:

$$|\psi\rangle \otimes |-\rangle \xrightarrow{U_{f_k}} \frac{1}{\sqrt{2^{n+1}}} \sum_{x=0}^{N-1} (-1)^{f_k(x)} \; |x\rangle \otimes |-\rangle. \tag{8.72}$$

We need to look more closely at the form of the output state. Remember that $f_k(x)$ is zero unless $x = k$. Thus each $|x\rangle$ in the above superposition has the same phase $(+1)$ as before except the state $|k\rangle$ which has a phase -1. Thus though we do not know what k is, this step is equivalent to *tagging* that particular state:

$$\sum_{x=0}^{N-1} (-1)^{f_k(x)} \; |x\rangle = |0\rangle + |1\rangle + \cdots - |k\rangle + \cdots + |N-1\rangle. \tag{8.73}$$

Algebraically, this step is equivalent to the action of the following *oracle operator* on the input register alone:

$$\hat{O} = \mathbb{1} - 2|k\rangle\langle k|, \tag{8.74}$$

since $|k\rangle\langle k|$ projects the state $|k\rangle$ out of the superposition. It helps to visualize this step, as well as the rest of the algorithm, by looking at what happens to the input state $|\psi\rangle$ in the Hilbert space $\mathcal{H}^{\otimes n}$. This space is spanned by the n unit vectors $\{|x\rangle\}$. Concentrate on the 2-d hyperplane spanned by the solution ket $|k\rangle$ and the vector $|\alpha\rangle$, a linear combination of all the other basis states, representing the hyperplane perpendicular to $|k\rangle$. You can visualize the input state $|\psi\rangle$ in this plane, as having a (small) component $\frac{1}{\sqrt{N}}$ along $|k\rangle$. The oracle operator $\hat{O}$ of Equation 8.74) reverses the sign of this component, performing a reflection in the hyperplane $|\alpha\rangle$ as shown in Figure 8.14.

Check out what this figure shows us. The initial uniform superposition is equally "far" from all the basis kets, including the target $|k\rangle$. In fact, it makes an angle

$$\theta = \sin^{-1} \frac{1}{\sqrt{N}} \tag{8.75}$$

with $|\alpha\rangle$ in this plane. The idea behind Grover's algorithm is to increase this

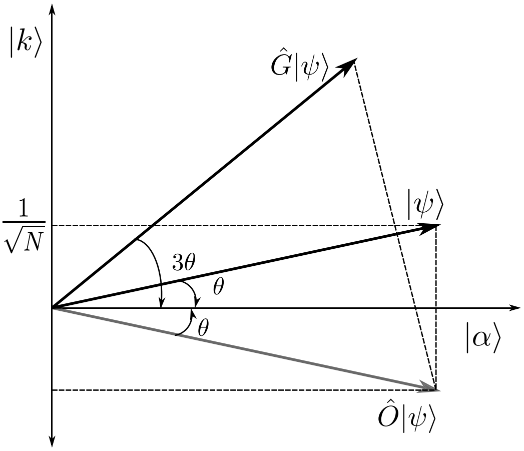


FIGURE 8.14: Geometric Visualization of the action of the Grover Iterate

angle to $\frac{\pi}{2}$ or as close to it as possible, by manipulating the phases of the state of the quantum computer.

Grover took a clue from the action of the operator $\hat{\mathcal{O}}$, and came up with another one $\hat{S}$, which implements a reflection in the plane of $|\psi\rangle$, which brings the input state closer to $|k\rangle$. By iterating this process a sufficient number of times, the input state $|\psi\rangle$ is rotated to $|k\rangle$. To see how this happens, let's construct $\hat{S}$:

$$\begin{aligned} \hat{S} &= \mathbb{1} - 2\big(\mathbb{1} - 2|\psi\rangle\langle\psi|\big) \\ &= 2|\psi\rangle\langle\psi| - \mathbb{1}. \end{aligned} \tag{8.76}$$

The action of the Grover iterate $\hat{G} = \hat{S}\hat{O}$ is to rotate the input state by an angle 3θ towards $|k\rangle$. This can be seen in the basis of vectors $|k\rangle$ and $|\alpha\rangle$:

$$\hat{G}|\psi\rangle = \cos 3\theta|\alpha\rangle + \sin 3\theta|k\rangle. \tag{8.77}$$

If this is repeated p times,

$$\hat{G}^p|\psi\rangle = \cos[(2p+1)\theta]|\alpha\rangle + \sin[(2p+1)\theta]|k\rangle. \tag{8.78}$$

Thus the iteration can stop when

$$\begin{aligned} (2p+1)\theta &\sim \frac{\pi}{2} \\ p &\sim \frac{1}{2\theta}\left(\frac{\pi}{2} - \theta\right). \end{aligned} \tag{8.79}$$

For large N, we have $\sin\theta = \frac{1}{\sqrt{N}} \simeq \theta$, so that the number of iterations required is given by

$$p = \frac{\pi}{4}\sqrt{N} - \frac{1}{2} = \mathcal{O}(\sqrt{N}). \tag{8.80}$$

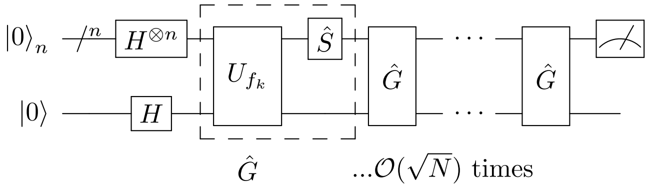


FIGURE 8.15: Circuit implementing Grover's algorithm

The circuit representation for this algorithm is given in Figure 8.15.

The process of rotating the superposition $|\psi\rangle$ towards the solution essentially works because of increasing the amplitude for $|k\rangle$, so this method goes under the name of "amplitude amplification".

Example 8.6.1. Let's look at the case $N = 4$, for 2-bit indices. The initial angle is given by $\sin\theta = \frac{1}{2} \implies \theta = \frac{\pi}{6}$. After a single iteration, the angle becomes $\frac{\pi}{2}$: thus a single run of the algorithm gives the answer.

Example 8.6.2. The deity who constructs the oracle in Grover's algorithm must give us a circuit implementing U_f for the checking the criterion. Let's say we have a 5-bit database and the 19^{th} entry is the search item. In binary, the index for the solution is $k = 18 = 10010$. The oracle output must be 1 for this input and 0 otherwise. It's easy to see that the circuit of Figure 8.16 will do the trick:

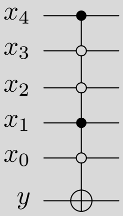


FIGURE 8.16: Construction of oracle for $k = 18$.

Example 8.6.3. We'll see how to construct the circuit for $\hat{S}$ (Equation 8.76).

$$\begin{aligned}\hat{S} &= -(\mathbb{1} - 2|\psi\rangle\langle\psi|) \\ &= -H^{\otimes n}\,(\mathbb{1} - 2|0\rangle\langle 0|)\,H^{\otimes n} \\ &= H^{\otimes n}\hat{P}H^{\otimes n}\end{aligned}$$

where the operator $\hat{P}$ leaves all basis states unchanged except $|0\rangle$, whose sign is flipped. (We can also ignore the overall negative sign.) This can be implemented by an $(n-1)$-fold 0-controlled Z gate. Since $Z = HXH$, we have the circuit of Figure 8.17 for $\hat{S}$.

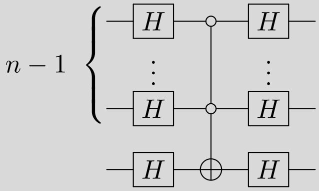


FIGURE 8.17: Construction of operator $\hat{S}$.

8.6.1 Extension to multiple solutions

A simple extension to this algorithm works when the search criterion has multiple solutions. The oracle then gives an answer "yes" whenever any one of the M possible solutions is input. The Hilbert space then has a "solution subspace" $\mathcal{M}$, spanned by M solution states. Let us denote by $|\beta\rangle$ the uniform superposition of all these vectors, and by $|\alpha\rangle$, its orthogonal complement.

$$|\beta\rangle = \frac{1}{\sqrt{M}} \sum_{x\in\mathcal{M}} |x\rangle, \tag{8.81}$$

$$|\alpha\rangle = \frac{1}{\sqrt{N-M}} \sum_{x\notin\mathcal{M}} |x\rangle. \tag{8.82}$$

In this situation, the input state is

$$|\psi\rangle = \sqrt{\frac{M}{N}}|\beta\rangle + \sqrt{\frac{N-M}{N}}|\alpha\rangle. \tag{8.83}$$

The angle θ is now given by

$$\sin\theta = \langle\psi|\beta\rangle = \sqrt{\frac{M}{N}}. \tag{8.84}$$

The operator $\hat{O}$ tags all of the M solutions with a '$-$' sign, and the Grover iterate is defined the same way as before. After p iterations we get the state

$$\hat{G}^p|\psi\rangle = \cos[(2p+1)\theta]|\alpha\rangle + \sin[(2p+1)\theta]|\beta\rangle, \tag{8.85}$$

and the number of iterations required is nearly

$$p \approx \frac{\pi}{4}\sqrt{\frac{N}{M}}. \tag{8.86}$$

You should check for yourself that this works out.

8.6.2 Quantum counting

A fallout of the multiple search algorithm is the counting algorithm. Before we know how many times to iterate, we need to know how many solutions M there are to the search criterion. Can we deduce a quantum algorithm for finding M given U_{f_k}? The solution found by Brassard et al. [15], is a combination of Grover's search and Shor's phase estimation algorithms. The key point here is to note that the number of solutions is related to the eigenvalues of the operator $\hat{G}$, which can also be expressed in the $|\alpha\rangle$-$|\beta\rangle$ basis as the 2-d matrix

$$\hat{G} = \begin{pmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{pmatrix} \text{ where } \sin\varphi = 2\frac{\sqrt{M(N-M)}}{N}. \tag{8.87}$$

The eigenvalues of this matrix are $e^{\pm i\varphi}$. We can therefore use the phase estimation algorithm to deduce φ. We need to feed the algorithm with an eigenstate of $\hat{G}$. Now you can see for yourself that $|\psi\rangle$ is a linear superposition of the two eigenstates of $\hat{G}$, so the circuit of Figure 8.18 will work to t bits of accuracy.

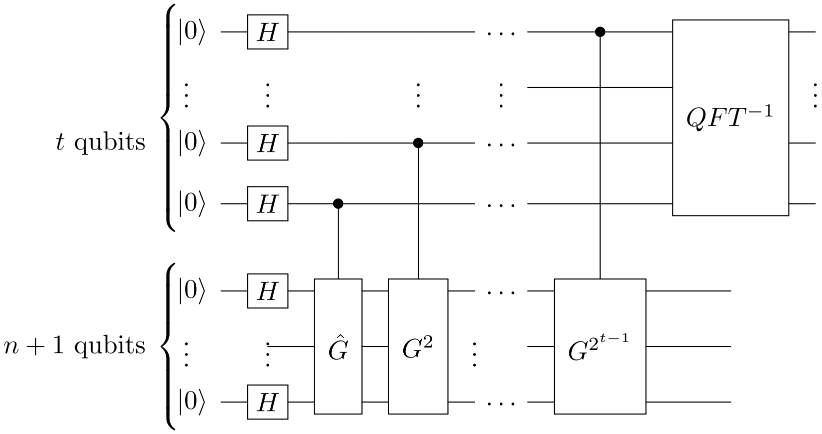


FIGURE 8.18: Circuit for quantum counting.

For Further Reading

The subject of quantum algorithms has rapidly evolved from its beginnings. Good references, apart from the original papers that may be found in the bibliography, are the excellent text book by Kaye, Laflamme, and Mosca [43] and the introductory text by Rieffel and Polak [58].

Problems

8.1. Some texts implement the quantum function evaluator as a "controlled-$\tilde{U}_f$" gate (Figure 8.19), where $\tilde{U}_f$ acts only on the lower register, and is defined by $\tilde{U}_f|y\rangle = |y \oplus f(x)\rangle$:

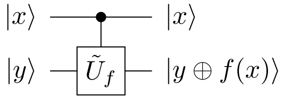


FIGURE 8.19: The quantum function evaluator as a controlled $\tilde{U}_f$ gate.

How is the action of this implementation different from the f-controlled NOT gate of Figure 7.15? Check by using standard basis states as well as superpositions as inputs.

8.2. Show that the phase kickback trick works because the input state in the bottom register is an eigenstate of the $\tilde{U}_f$ operator for the Deutsch algorithm.

8.3. Deutsch's original version of his algorithm used $|0\rangle$ as the input to the bottom register instead of $|0\rangle - |1\rangle$. Show that in this case you obtain the correct answer with probability $3/4$. Also show that the algorithm has probability $1/2$ of succeeding.

8.4. Prove the shift-invariance property of the Fourier transform, i.e., show that

$$\hat{\mathcal{F}}|x+k\rangle = e^{i\theta}\hat{\mathcal{F}}|x\rangle \tag{8.88}$$

for some θ. Find θ in terms of k.

8.5. For the operator R_d of Equation 8.45, give a construction for the controlled R_d gate using CNOT and single-qubit gates.

8.6. Find the eigenvalues and eigenvectors of the matrix R_d. What can you say about the commutators (i) $[R_d, X]$ (ii) $[R_d, Y]$ (iii) $[R_d, Z]$ (iv) $[R_d, R_d']$?

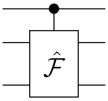
8.7. Work out a circuit that calculates the inverse quantum Fourier transform.

8.8. Consider a periodic function $f(x+r) = f(x)$ for $0 \leq x < N$ where N is an integer multiple of r. Suppose you are given a unitary operator U_y that performs the transformation $U_y|f(x)\rangle = |f(x+y)\rangle$. Show that the state

$$|\tilde{f}(k)\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} e^{-2\pi ikx/N} |f(x)\rangle \tag{8.89}$$

is an eigenvector of U_y. Calculate the corresponding eigenvalue.

8.9. Compute the output of the controlled-QFT gate shown in the figure if the input is $H^{\otimes 3}|x\rangle$.



8.10. On examining the period finding algorithm, we can find a relationship with the phase-estimation algorithm. On applying the oracle, we get

$$\frac{1}{\sqrt{N}} \sum |x\rangle|0\rangle \xrightarrow{U_f} \frac{1}{\sqrt{N}} \sum |f(x)\rangle.$$

Express $|f(x)\rangle$ in terms of its Fourier transform, $|\tilde{f}(k)\rangle$. Invert this expression and show that $|\tilde{f}(k)\rangle$ are of the same form as Equation 8.89 of Problem 8.8. Now show that the period finding algorithm is the phase estimation for the operator U_y defined there.

8.11. Apply the quantum phase estimation algorithm to the following cases and obtain the results:

(a) $U = X, \quad |u\rangle = |-\rangle, \quad t = 2,$

(b) $U = R_d, \quad |u\rangle = |1\rangle, \quad t = d+1.$

Part IV

Quantum Information

Chapter 9

Information and Communication

The most successful practical applications of quantum mechanics in information theory have been in the field of communication. This is a very old field that has taken inputs from engineering, pure mathematics and computer science apart from physics. It is only natural that application of quantum techniques here should be among the first to be considered and implemented. Modern technology in optical communication using laser light and optical fiber cables is sufficiently advanced that it can quickly be adapted to using photons as the quantum carriers of information.

In analyzing information communication, we normally consider the following areas:

1. Coding: representing information accurately in terms of physical variables and the removal of redundancy leading to more efficiency, which is known as *compression*;

2. Transmission: the information-carrying capacity of a channel and the possible errors introduced into the data, and how to analyze and correct them; and

3. Secure communication: including data encryption techniques, especially sharing of secret keys for encryption.

Before we look into the nature and characterization of information in quantum systems, we need some terminology commonly used in this subject. The classic paradigm of information and communication is indicated in the cartoon in Figure 9.1.

Two parties that may be in separate locations (we call them Alice and Bob after the tradition in communication theory) need to communicate some data. The term **data** refers to information converted into a form suitable for the physical protocol being used for transfer. Data is produced from information by a process called **encoding**. This process is essentially a mathematical function executed by a computing machine. At this stage, we are concerned with the quantification of information present in the data, the efficiency of encoding and how much the data can be compressed. If the security of the data is of importance, then this is the stage where secrecy is built into the message, a process called **encryption**.

Alice, the sender, then transmits the encoded data by physical means known as the **channel**. At this stage, we are concerned about the efficiency of

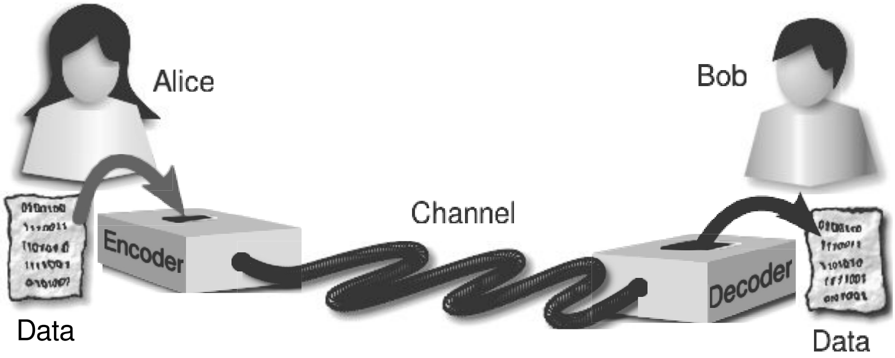


FIGURE 9.1: Communication of information

the channel, quantified by the rate at which data can be transmitted by the channel. Another important factor at this stage is noise. Data could get corrupted by various means and an error-correcting scheme like those we looked at in Chapter 10 has to be built into the communication protocol. If the data needs to be securely transmitted, then at this stage an eavesdropper (Eve) can tap into the channel and check how much it can be compromised.

When the data reaches its destination with Bob, it needs to be **decoded** to be readable by Bob. This process is essentially the reverse of encoding, and the efficiency of the whole protocol can be computed at this stage. The security of the protocol can also be checked if Alice and Bob now compare some of their data through other means.

Quantum data processing can help us with making this process more efficient as well as more secure, as we will see in this chapter. The chief properties of quantum systems that will be exploited here are entanglement and the indistinguishability of non-orthogonal states.

9.1 Entanglement as a Resource

As we have seen in Chapter 4, multi-qubit systems can exist in correlated states known as entangled states. In computing algorithms, entanglement is implicit, but nowhere is it more dramatically useful than in protocols for communication, where quantum correlations are exploited for security of communication as well as for coding efficiency. This quantum correlation shared by spatially separated parties is used as a communication resource. The term **ebit** has been coined to quantify this resource. An ebit can be thought of as the amount of entanglement in a Bell state.

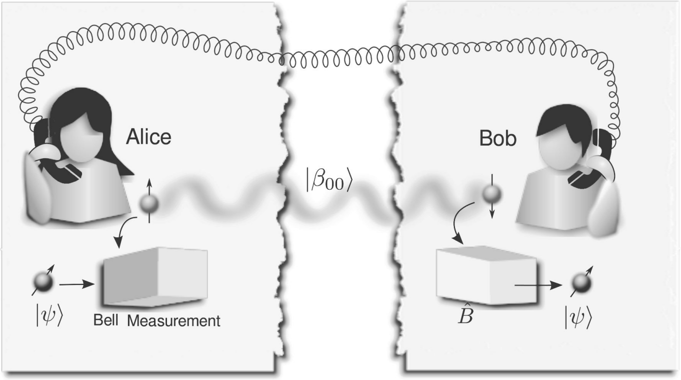


FIGURE 9.2: Quantum Teleportation

9.1.1 Teleportation

As an illustration of the power of entanglement as a resource, we examine the rather dramatically titled protocol of quantum state teleportation. This idea was first introduced by Bennett in 1993 [6]. The teleportation problem is the following: Alice needs to transfer to Bob (at a distant location) an unknown qubit $|\psi\rangle$ generically denoted by $\alpha|0\rangle + \beta|1\rangle$. The key point here is that Alice does not know what α and β are. Quantum channels are not available for use, so she cannot simply transmit the qubit to Bob. The unknown state of the qubit cannot be determined since a measurement would destroy the state. Multiple measurements need to be performed on identical copies of the state in order to estimate α and β, but Alice has only one copy, and the no-cloning theorem forbids her from making more copies.

Prior to the process, we assume that Alice and Bob share an entangled pair of qubits in the state $|\beta_{00}\rangle$. The protocol, illustrated in Figure 9.2, works as follows: Alice first makes a Bell measurement on the two qubits in her possession (one unknown qubit and the other entangled with Bob's qubit). Refer to Figure 7.12 of Chapter 7 for the circuit equivalent to this process.

The results of her measurements are two classical bits of information, which Alice now transmits to Bob, through a standard classical channel. Then Bob can basically retrieve the quantum state $|\psi\rangle$ by performing certain predetermined operations $\hat{B}$ on his qubit, that depend on the result of Alice's measurements. We can see how this works by representing the process as a circuit (Example 7.2) and working through it. Bell measurement involves transforming the two qubits into the Bell basis and then measuring them. The state of the three particles just before Alice measures her two qubits is

$$\begin{aligned} |\phi\rangle \quad &= \quad \frac{1}{4}|00\rangle\big[\alpha|0\rangle + \beta|1\rangle\big] + \frac{1}{4}|01\rangle\big[\alpha|1\rangle + \beta|0\rangle\big] \\ &\quad + \frac{1}{4}|10\rangle\big[\alpha|0\rangle - \beta|1\rangle\big] + \frac{1}{4}|11\rangle\big[\alpha|1\rangle - \beta|0\rangle\big] \end{aligned} \tag{9.1}$$

Upon Alice's measurement, all three qubits collapse to one of the states in Table 7.1. Thus to retrieve $|\psi\rangle$, Bob must perform one of the set of conditional operations in Table 9.1.

TABLE 9.1: Bob's conditional operations in the teleportation protocol.

Alice transmits	Bob performs
00	$\hat{B} = \mathbb{1}$ (Identity)
01	$\hat{B} = X$
10	$\hat{B} = Z$
11	$\hat{B} = ZX$

The entire protocol can be represented by the circuit in Figure 9.3.

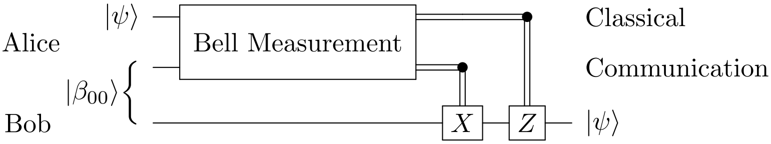


FIGURE 9.3: Circuit for teleportation.

We've worked through this circuit in Example 7.2, and you should have no doubts that the state $|\psi\rangle$, which was initially with Alice, is finally in Bob's line. This process uses up the entangled pair, which is why we regard entanglement as a resource.

9.1.2 How teleportation does not imply faster-than-light communication

A niggling question (which certainly worried Einstein as recorded in the EPR paper [31]) would be how the information contained in $|\psi\rangle$ was "instantaneously" transferred from Alice to Bob when Alice measured her qubits. The key point here is that no such signaling that is faster than light (thereby violating the special theory of relativity) is in fact occurring. Until Bob actually knows what the outcome of Alice's measurements were, he does not know that he is in possession of the qubit $|\psi\rangle$. Thus, information is transferred only when Alice conveys to him her measurement outcomes, and in this scheme, she does not signal faster than light, but is in fact using conventional (classical) methods of communication. In fact, the processes adopted in this typical protocol are an example of "*local operations and classical communication*" or **LOCC**, which is one of the key phrases in quantum information theory.

Box 9.1: No Signaling Theorem

The fact that quantum mechanics does not allow distant parties to exchange information instantaneously using the non-local correlations of entanglement, can be proved neatly using the density operator formalism. Suppose Alice and Bob share a state

$$\rho^{AB} = \sum_{i,j} p_{ij} |i\rangle^A |j\rangle^B$$

that may be entangled. Suppose Alice performs a measurement on her system, characterized by generalized measurement operators M_m. How does this affect the state of Bob's system? Bob's new density matrix is

$$\begin{aligned}
\rho'^B &= \mathrm{Tr}_A \left[\sum_m (M_m \otimes \mathbb{1}) \rho^{AB} (M_m^\dagger \otimes \mathbb{1}) \right] \\
&= \sum_m \mathrm{Tr}_A \left[(M_m \otimes \mathbb{1}) \rho^{AB} (M_m^\dagger \otimes \mathbb{1}) \right] \\
&= \sum_m \mathrm{Tr}_A \left[(M_m^\dagger M_m \otimes \mathbb{1}) \rho^{AB} \right] \\
&= \mathrm{Tr}_A \left[\sum_m (M_m^\dagger M_m \otimes \mathbb{1}) \rho^{AB} \right] \\
&= \mathrm{Tr}_A \rho^{AB} \\
&= \rho^B .
\end{aligned}$$

Thus it is not possible to affect Bob's state by any local operation performed by Alice: Bob's knowledge cannot be changed — information cannot be conveyed — by Alice through the non-local correlations of entangled states.

9.1.3 How teleportation does not imply cloning

Another common misconception for a beginner in quantum mechanics is that teleportation looks as if the state $|\psi\rangle$ is copied out from Alice's location to Bob's. A little consideration will show that in fact this is not happening. The moment Alice measures her qubits, the state $|\psi\rangle$ ceases to exist on her end. She only has two classical bits with her. The unknown state with its implicit α and β coefficients is completely transferred to Bob. The state $|\psi\rangle$ exists only in one location: either at Alice's end or at Bob's, and is NOT cloned at any point.

9.2 Quantum Dense Coding

An interesting aspect of quantum information transfer is how one can actually transfer two classical bits of information while physically transmitting only one qubit. This process seems to involve compressing two bits into one qubit and is accordingly called *dense coding*. The key to the process is the use of entanglement. This protocol preceded and inspired the teleportation protocol discussed above [10]. So our friends Alice and Bob enter the picture with their shared Bell state, which they are going to use as a resource to communicate two bits of information between them.

The trick is fairly simple. Suppose Alice and Bob share the Bell state $|\beta_{00}\rangle$. Alice performs a local operation on her piece of the entangled pair depending on the two-bit number she wishes to communicate, and then transfers the qubit over an appropriate quantum channel to Bob. Bob then measures both qubits in the Bell basis to obtain the two-bit number. The local operation $\hat{A}$ that Alice performs is according to Table 9.2.

TABLE 9.2: Operations for super-dense coding.

Number	Operation
00	$\hat{A} = \mathbb{1}$
01	$\hat{A} = \hat{X}$
10	$\hat{A} = \hat{Z}$
11	$\hat{A} = \hat{X}\hat{Z}$

Let's check how this works on an example: suppose Alice wishes to communicate the number 2 or 10 in binary. The sequence of operations undergone by the Bell pair is then as follows:

$$|\beta_{00}\rangle = \frac{1}{\sqrt{2}}[|00\rangle + |11\rangle] \xrightarrow{\hat{Z}^A} \frac{1}{\sqrt{2}}[|00\rangle - |11\rangle] \xrightarrow{\text{Bell basis change}} |10\rangle. \quad (9.2)$$

You can verify the last step by performing the operations for the Bell measurement explicitly as a CNOT and then an H on the first qubit.

Exercise 9.1. Show how the above dense coding protocol works if the entangled state shared by Alice and Bob was $|\beta_{11}\rangle = \frac{1}{\sqrt{2}}[|10\rangle - |01\rangle]$.

9.3 Quantum Cryptography

The most spectacular successes of quantum information processing techniques have been in the field of cryptography, the science of secret message exchange.[1] The reason why Shor's algorithm shot into prominence and major players started funding quantum computing research was the challenge it offered to currently trusted schemes of data encryption, particularly the RSA scheme that is the basis of almost all current public encryption systems, your online banking transactions or purchases for instance!

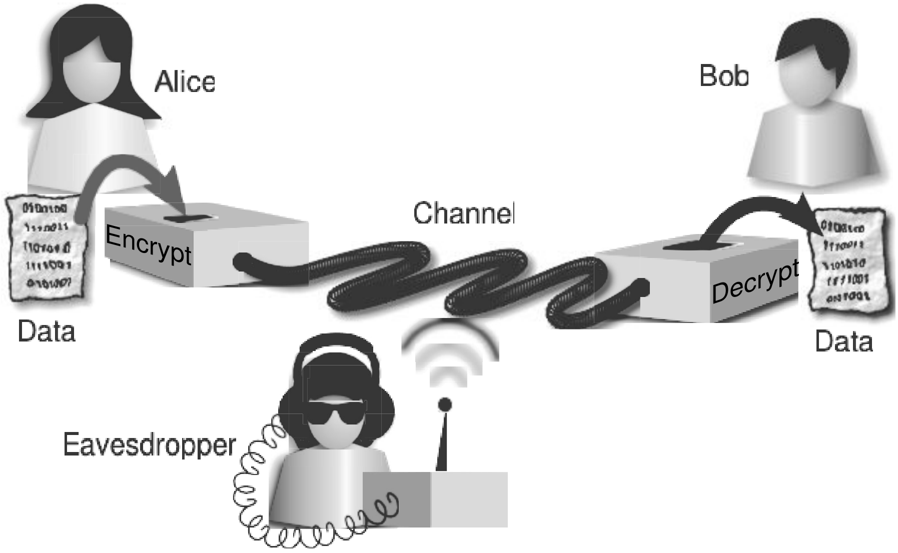


FIGURE 9.4: Communication scenario for cryptography.

In this section we will provide a quick birds-eye view of major cryptographic paradigms and where quantum information processing steps in to make things better. For a delightful survey of the history and current trends in cryptography I urge you to read the book by Simon Singh [66]. The progress in quantum cryptography is comprehensively dealt with in the review article by Gisin et. al. [38].

[1] The word "cryptography" is derived from the Greek language: crypto="secret," graphy="writing." It is actually one part of the science of "cryptology," the second part being "cryptanalysis," which is the art of decoding an encryption. These two go hand-in-hand: to test the success of any cryptographic scheme a thorough cryptanalysis is important.

9.3.1 Basic cryptographic paradigms

Almost ever since mankind used language for communication, need was felt for secrecy in that communication, as a protection of personal or national interests. The basic scheme (Figure 9.4) is the conversion of a natural language into a secret form, i.e., encryption, before transmission, and this needs to be tested against different eavesdropping techniques. Several interesting cryptographic schemes have evolved as our mathematical and logical prowess increased. For instance, many of us may have played as children by exchanging secret notes in which the text was encoded by replacing each letter by another shifted down the alphabet by a few letters. This in fact was an ancient cipher system attributed to Julius Caesar! The receiver then decodes the message by shifting the letters back by the same amount.

Example 9.3.1. The Caesar Cipher: suppose you decide to encode by shifting each alphabet by 5 letters:

Plain:	A	B	C	D	E	F	G	H	I	J	...	Y	Z
Cipher:	F	G	H	I	J	K	L	M	N	O	...	D	E

then the message “THIS IS A SECRET” would be encoded as “YMNX NX FJHWT.” The sender and receiver both agree as to what scheme of encoding they’ll use. The danger in such messaging is that if the message is intercepted, then a clever cryptanalyst can figure out the scheme used and easily translate any further messages sent by the same scheme. One can try to devise more complicated translation schemes, but as long as they are one-to-one, statistical methods such as the average frequency of letters in the English language may be used to break the code.

The basic paradigm for any secret communication has two requisites:

1. An eavesdropper should not be able to decrypt the message
2. The sender should not be impersonated.

The process of encryption is basically a mathematical transformation E on the input message m, which converts the plain text into a coded *ciphertext* c. The set of symbols used for the cypher is known as the *alphabet.* The physical analog is placing the message in a box that is then locked. The locked box is then transported to Bob, who opens the box, i.e., decodes the message, and obtains the plain text again. In this raw form of the protocol, security is minimal since the message can always be intercepted by an eavesdropper (Eve), who can then try to break the code (or recreate the key). The only way this threat can be met is that each time Alice wishes to send a message, she uses a new box (or a new algorithm for encryption) and that is wasteful. Instead, what she opts for is to change the lock and therefore the *key* K_A used for locking the box. Bob then unlocks using his key K_B, which has to be the correct one for the lock Alice used. The problem of keeping the message secret now reduces to keeping the keys secure.

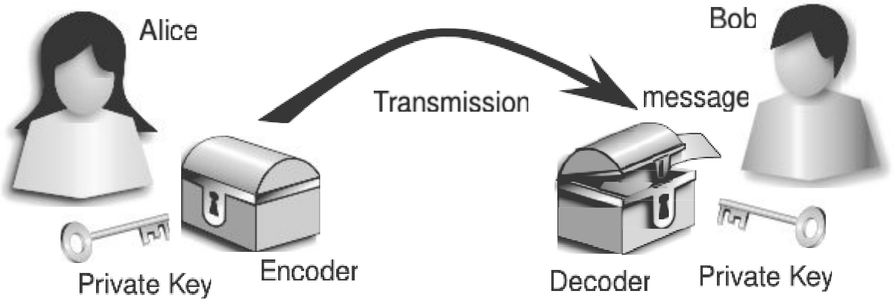


FIGURE 9.5: Private key cryptography.

Mathematically, we represent the process of cryptography by

$$m \rightarrow E(m, K_A) = c \text{ (encryption)} \tag{9.3}$$

$$c \rightarrow D(c, K_B) = m \text{ (decryption).} \tag{9.4}$$

The functions E and D need to be inverses of each other in this sense:

$$D(E) = \mathbb{1}. \tag{9.5}$$

The encrypting function can be kept simple, so as to be computationally efficient, and can be publicly known. This is the modern principle of cryptography, sometimes known as **Kerckhoff's principle**. The weak point of an encryption system should be easily changed if it falls into the enemy's hands. The choice then is for the key system, whether the encoding key is kept secret or not. This results in two sets of schemes:

1. Symmetric or private key cryptography: Alice and Bob share the *same* key K (Figure 9.5). This is like the box with a lock, whose key is shared by both parties. The sharing must be done in an efficient and secret way. The catch is in this step. If A and B are far separated, how can one transmit the key to the other in a secure way? This is the problem of **secure key distribution**.

2. Public key cryptography: here the same key is not used by both parties. The sender uses a public or insecure key K_A to encrypt the message (Figure 9.6). The decryption process is achieved by a private, secure keyK_B. This process is like a ballot box that is locked and given to the sender, who posts the message in it. The receiver alone can unlock it with his secret key. Here the E and D processes are asymmetric, and the problem of distribution of keys doesn't arise. The security of the protocol lies in the difficulty of operating D without the knowledge of K_B.

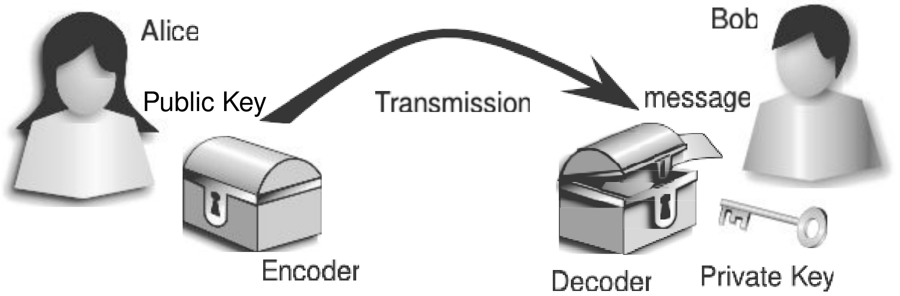


FIGURE 9.6: Public key cryptography

Due to the difficulty in sharing truly secure keys, especially when a large number of parties are involved, modern cryptographic schemes are usually of the second kind. One of the most widely used protocols for public-key encryption is a two-step process due to Diffie and Hellman [28], and by Merkle [47]. Known as the D–H protocol, Alice and Bob each have a private key denoted L and a public key denoted K.

$$\begin{aligned} \text{Encryption (Alice): } & c_1 = E(m, L_A); \quad c = E(c_1, K_B) \\ \text{Decryption (Bob): } & m_1 = D(c, L_B); \quad m = D(m_1, K_A). \end{aligned} \tag{9.6}$$

Exercise 9.2. Show that in the two-way public key cryptosystem of Equation 9.6, E and D are indeed inverses of each other.

Example 9.3.2. Private key cryptography: the Vernam cipher or one-time pad.

A and B agree on a common encryption system and share a common secret key K. One example of such an encryption is encoding the message in N symbols and performing a bitwise addition mod N with the key. The inverse is performed using the same key.

$$c = m + k \mod N, \qquad m = c - K \mod N.$$

- The key is a one-time use only. This is because it can be easily reconstructed from the cipher if it is intercepted.
- The advantage of this technique is that the process is computationally simple.
- C. Shannon has proved that this system is truly unbreakable as long as the key is secret and is of the same length as the message.

Example 9.3.3. The popular RSA encryption scheme [59], invented in 1978 by Rivest, Shamir, and Adleman at MIT, is a public key system based on the prime factorization of a large number N. While the system is not truly unbreakable, its strength lies in the fact that the private key is computationally hard to generate though the generation of the public key is easy.

The steps B follows to generate his public and private keys are as follows:

1. Randomly selects two large primes p, q.
2. Computes $N = pq$.
3. Randomly selects a small odd a coprime to $(p-1)(q-1) = r$.
4. Picks b such that $ab \mod r = 1$.
5. Public key: N and a
   Private key: N and b.

$$\begin{aligned} \text{Encryption: } E : c &= m^a \mod N \\ \text{Decryption: } D : m &= c^b \mod N. \end{aligned}$$

How are E and D inverses of each other? A little number theory comes into play here:

$$D : m^{ab} \mod N.$$

Now $ab = 1 + kr = 1 + k(p-1)(q-1)$ for some integer k. If m doesn't divide q, (which is true since q is prime,) then

$$m^{ab} = m \cdot (m^{q-1})^{k(p-1)} \mod q.$$

By Fermat's little theorem, $m^{(q-1)} \mod q = 1$. So

$$m^{ab} = m \cdot 1^{k(p-1)} \mod q = m \mod q.$$

By similar reasoning, we also have

$$m^{ab} = m \mod p.$$

A result known as the Chinese remainder theorem guarantees that if this is so then

$$m^{ab} = m \mod pq = m \mod N.$$

While public key cryptosystems are more practicable, they are not truly unbreakable. It is this vulnerability that has been shown up by Shor's factorization algorithm. Thus the emphasis is now on more efficient and secure private key systems. Here a private key needs be generated in advance of sending the message. It is at least as long as the number of characters in the message, and a new key needs to be generated for each use. The keys need to be distributed over large distances in a secure fashion and this is what quantum cryptography has been primarily about.

9.3.2 Security of cryptosystems: possible attacks

The vulnerability of any cryptosystem needs to be subjected to stringent tests before it can be implemented. In fact the very growth of new and efficient cryptographic schemes depends a lot on the input of cryptanalysts, who make a thorough study of possible loopholes and susceptibility to attacks. A study of possible methods that endanger a system would require a book of its own. Possible attacks on a cryptosystem are

1. Decoding: the most obvious of all — an eavesdropper intercepts the message and solves for the decoding keys
2. Eavesdropping (the message is intercepted and decoded) if detected, means that the channel is insecure and needs protection or needs to be dropped altogether
3. Man-in-the-middle or impersonation: an eavesdropper having access to the channel impersonates the sender and thus gets information about the decoding scheme, or else foils the communication. This is especially true when there is no means of authenticating the sender
4. Denial of service: the eavesdropper is able to clog the communication channel or even cut it off physically and prevent the transmission of messages.
5. Other attacks specific to the hardware and protocols being used.

9.4 Quantum Key Distribution

Quantum key distribution is potentially secure because of the fundamental properties of the quantum states used. The first schemes of quantum key distribution relied on the indistinguishability of non-orthogonal states and the no-cloning principle. Another reason why quantum key distribution has been such a resounding success is that the protocols are feasible and immediately implementable using available optical technology. Quanta of light, photons, are used to carry qubits. The two basis states are implemented by the two orthogonal states of polarization of the light. Various bases can be used to represent the polarization. Linearly polarized light in different basis states can be easily produced by passing light through a polarizer with pass axis oriented along different directions.

𝟙-basis:

$$|0\rangle : \quad \text{horizontally polarized} \; : \quad |\leftrightarrow\rangle \tag{9.7a}$$

$$|1\rangle : \quad \text{vertically polarized} \; : \quad |\updownarrow\rangle. \tag{9.7b}$$

H-basis:

$$|+\rangle = H|0\rangle : \quad \text{polarized at } +45^\circ : \quad |\nearrow\!\!\!\!\swarrow\rangle \tag{9.8a}$$

$$|-\rangle = H|1\rangle : \quad \text{polarized at } -45^\circ : \quad |\nwarrow\!\!\!\!\searrow\rangle. \tag{9.8b}$$

The circular polarization basis is also sometimes used as it is easily produced by using quarter-wave plates in conjunction with polarizers.

Y-basis:

$$|i\rangle = S|0\rangle : \quad \text{right circular polarized} : \quad |\circlearrowright\rangle \tag{9.9a}$$

$$|-i\rangle = S|1\rangle : \quad \text{left circular polarized} : \quad |\circlearrowleft\rangle. \tag{9.9b}$$

These states are indicated on the Bloch sphere in Figure 9.7.

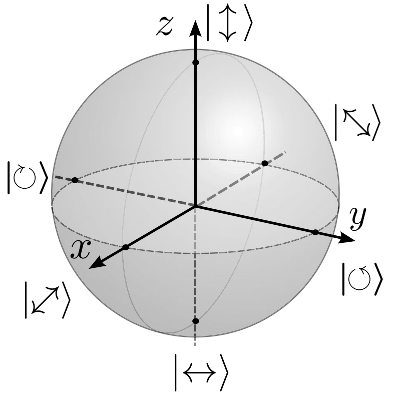

z
$|\updownarrow\rangle$
$|\nwarrow\!\!\!\!\searrow\rangle$
$|\circlearrowright\rangle$
y
x
$|\circlearrowleft\rangle$
$|\nearrow\!\!\!\!\swarrow\rangle$
$|\leftrightarrow\rangle$

FIGURE 9.7: Different photon polarization states indicated on the Bloch sphere.

If a bit is encoded in a photon prepared randomly in one of the states of Equations 9.7 and 9.8, can you find out which bit I have, without knowing my preparation basis? The answer is, not with certainty. The best you can do is to measure the photon in one of the 4 bases, chosen at random. What are the chances that you pick the right one?

Say I prepared a $|+\rangle$, which is the bit 1 encoded in the H basis. The probability of your choosing the right basis for measuring is $1/2$. If you chose the wrong basis, then the probability of your measuring a 1 is $1/4$. If I have a whole string of n bits encoded in this fashion, the probability that you guess right will be $(1/4)^n$ which becomes exponentially tinier as n increases!

Protocols for secure sharing of a random bit string between two parties rely on this property of encoding. We will review a few of them here to show you how it works.

9.4.1 BB84 protocol

Due to C. Bennett and G. Brassard in 1984 [9], this protocol seeks to generate a perfectly random bit string that is shared by Alice and Bob. The beauty of the method is that the bit string does not exist until Bob measures the qubits Alice has transmitted to him. Thus the security of the shared string is guaranteed. The bits are randomly encoded either in the computational basis ($\mathbb{1}$) or in the H basis. The steps followed are:

1. Alice produces a random bit string s_A (for instance, by making quantum measurements on an unpolarized stream of qubits) of length l.
2. She uses another random bit sequence m_A to choose which polarization state to encode each bit in: $\mathbb{1}$ if $(m_A)_i = 0$ and the H basis if $(m_A)_i = 1$.
3. This encoded stream of photons is transmitted to Bob across a quantum channel. We label the state of the i^{th} photon by $|\phi_i\rangle$.
4. Bob now uses a random bit string m_B to choose a basis for measuring each photon in this stream as it comes to him. He then has a string of measurement outcomes s_B.
5. After the measurements have been made, Alice announces her string m_A over a public (insecure) channel.
6. Bob discusses with her and they discard those bits of s_A and s_B for which the measuring bases **do not** match, i.e., those bit positions in m_A that do not match with m_B.
7. The remaining bits, corresponding to the matching places, form a possible shared key. On an average, there will be half the original number of bits in this set. a

The probability of Bob choosing the same basis as Alice is one half, so they must start out with a string at least twice as long as the intended key. The security of this method hinges on the inability to unambiguously distinguish bits encoded in non-orthogonal bases.

Example 9.4.1. An example of the BB84 protocol is shown below, with the shared key bits highlighted. Where the measurement bases are not the same, the state measured by Bob is left blank, as it could randomly be $|0\rangle$ or $|1\rangle$.

Index	1	2	3	4	5	6	7	8	9	10	11	12	13	14
s_A	1	0	0	0	1	1	1	0	1	0	0	0	1	1
m_A	0	1	1	0	1	1	1	0	1	1	0	0	0	1
$\|\phi_i\rangle$	↔	⤢	⤢	↕	⤡	⤢	⤡	↕	⤡	↔	↕	↕	⤢	⤡
m_B	0	0	0	1	1	0	1	0	1	0	0	0	1	1
s_B	1	0	1	0	1	1	1	0	1	1	0	0	1	1

Candidate key $k = 11101001$

What if there is an eavesdropper on the channel? Suppose Eve gains access to the qubits in the quantum channel. She cannot copy the qubits and then send them on their way to Bob, since the no-cloning theorem ensures she will not have faithful copies. She can, however, measure the qubits in her own choice of bases and then send them onward. In this situation, she has a 50% chance of choosing the same basis as Alice. When Bob measures the qubits again, he has a 50% chance of having chosen the same basis as Eve, so that on the whole he has only a 25% chance of agreeing with Alice's choice! But how does he discover that the channel security has been compromised?

Alice and Bob decide to **test** this, by agreeing to compare a fraction of their shared bit string. They can do this over a public channel, and if they discover up to 25% mismatch then they know that the channel is suspect, and they will not use it for their communication.

The protocol can be divided into three phases: first, the sending of the bit-stream encoded in a quantum channel and the measurements made by Bob; second, the public discussion of the data they obtain and third, the sifting, testing and authentication of their data. We will discuss the last two a little later.

9.4.2 BB92 protocol

The BB84 protocol was further refined in 1992 to use just two different encoding states instead of four. The only two states Alice uses are $|\updownarrow\rangle$ and $|⤢\rangle$. This is sufficient, since they are not orthogonal and cannot be reliably distinguished by the eavesdropped. The key steps are as follows:

1. Alice creates a random bit string s_A,

2. She encodes a string 0's in photons polarized randomly in the $\mathbb{1}$ or H basis according to the bits in s_A.

3. Bob chooses a random string s_B according to the bits in which he chooses the basis $\mathbb{1}$ or H to measure the photons. The measurement results form a string m_B.

4. Now Bob creates a string $k \in s_B$ keeping only those the bits positions where $m_B = 1$.

5. Bob publicly declares those bit positions, so that Alice can select those bits from s_A.

6. Since Alice encoded only 0, Bob can measure a 1 only when their bases were **not** the same. Therefore, if Bob's bit is k_i then Alice's bit in the same position will be $1 - k_i$. Thus after Alice takes the complement of her bits, both have a shared key k.

The reason this works is as follows: Alice sending a $|\updownarrow\rangle$ means a 0 and a $|\nearrow\!\!\!\!\swarrow\rangle$ means a 1. Now Bob randomly decides to use the $\mathbb{1}$ or H basis. But notice that when he has used the computational basis, and if Alice had sent a $|\updownarrow\rangle$ then he would always gets a 0, and if she'd sent $|\nearrow\!\!\!\!\swarrow\rangle$ he can get 0 or 1 with probability 1/2. Thus the only places where Bob gets a 1 will be when Alice and Bob have complementary bits in their random string s.

Example 9.4.2. An example of the BB92 protocol is shown below.

Index	1	2	3	4	5	6	7	8	9	10	11	12
s_A	1	1	0	1	0	0	0	1	1	1	0	1
$\vert\psi_i\rangle$	$\nearrow\!\!\!\!\swarrow$	$\nearrow\!\!\!\!\swarrow$	$\updownarrow$	$\nearrow\!\!\!\!\swarrow$	$\updownarrow$	$\updownarrow$	$\updownarrow$	$\nearrow\!\!\!\!\swarrow$	$\nearrow\!\!\!\!\swarrow$	$\nearrow\!\!\!\!\swarrow$	$\updownarrow$	$\nearrow\!\!\!\!\swarrow$
s_B	1	1	1	0	1	0	1	0	0	0	1	0
m_B	0	0	0	1	1	0	0	1	1	0	1	1
K_A				1	0			1	1		0	1
K_B				0	1			0	0		1	0

Shared key is $k = 010010$.

In this protocol too, the effect of the presence of an eavesdropper in the channel is the same as for the BB84. Alice and Bob will again have to sacrifice a few of their shared bits to verify the security of the channel.

9.4.3 E91: QKD using entangled states

Here is another variant of the QKD protocol, due to Ekert [32], which makes use of correlated quantum pairs as a resource shared between Alice and Bob. For each key, Bob generates entangled photons and sends one to Alice.[2] Locally each performs a measurement randomly, according to random bit strings m_A and m_B respectively, in the $\mathbb{1}$ or the H basis. These bit strings

[2] It doesn't matter who generates the pair. It could also be generated by a third party and sent to both of them. The quantum channel is used for this purpose.

are shared over a public channel, and Alice and Bob compare them to see which bits match. The measured values of those bit positions are retained as the shared key.

The point is that when m_A and m_B match, the measurement results, though random, are perfectly correlated, while when they don't match, Alice and Bob get the same result only 50% of the time.

The drawback of this scheme is that Alice and Bob would have to verify that their photons retained their entanglement when the key was being generated. To do this, they would have to perform an additional exercise of, say making sure Bell's inequality was violated. (For instance, each of them could measure their photons in three different bases and share the values.)

Example 9.4.3. An example of the entangled QKD scheme: suppose Alice and Bob share a huge supply of qubits in the state $|\beta_{00}\rangle$.

Index	1	2	3	4	5	6	7	8	9	10	11	12
m_A	0	1	0	1	0	0	0	1	1	1	0	0
m_B	1	1	1	0	1	0	1	0	0	0	1	0
K		0				0						1

The presence of an eavesdropper Eve is detected the same way as for the BB84 scheme. The list of possible secure key-distribution schemes is quite long, and you are invited to contribute to it!

9.5 Information Reconciliation and Privacy Amplification

The sifting procedure in QKD protocols is to ensure the degree of security of the channel. The presence of an eavesdropper is detected by errors above a certain tolerance margin, say 20%. However, natural errors in the channel could also cause discrepancies in the shared key. To remove these, and to ensure further security on the shared key, two classical procedures known as *information reconciliation* (a form of error correction) and *privacy amplification* are carried out.

The basic idea of information reconciliation is to perform a parity check on a subset of the key, compare, and correct. At the two-bit level, parity is just an XOR. So Alice could randomly select two bits out of k_A, announce their positions and XOR to Bob. He then compares the parity of the same bits in k_B. If they do not match these bits are discarded. If they do then they decide to discard the second bit. This ensures that Eve does not learn anything more about their key from their discussion.

The more sophisticated version generalizes this process, as first described in 1992 by Bennett et al. [8]. They proceed in several iterations of essentially the same process, but first dividing their keys into predetermined *blocks* and checking the parity of the block. If the parity doesn't match then they recursively bisect their blocks to detect the location of the error and discard it. To ensure that Eve doesn't learn anything more from their parity discussions (which happen in public), they discard the last bit of each block whose parity is disclosed. This process is repeated many times with increasing block sizes, until eventually the two keys are ensured to be reconciled with a large probability.

At the end of information reconciliation, Alice and Bob have identical keys but whose privacy has been compromised by all the public discussions. To undo this effect, they resort to privacy amplification. To do this they select something called a *universal hash function* to encode their strings. There are many such functions that provide various bounds for the amount of information Eve can gain. One such is to select random subsets of their strings and to retain their parity bits for a new key.

In any case, both these steps amount to classical error correction and coding, and will not be dealt with at greater depth in this book.

It is clear from this discussion that depending on the degree of privacy they choose to have, the initial string length must be fairly large, of the order of 4 times the length of the desired key.

Problems

9.1. How would the teleportation protocol change if the entangled state shared by Alice and Bob was any of the other Bell states: $|\beta_{01}\rangle, |\beta_{10}\rangle$, or $|\beta_{11}\rangle$?

9.2. Consider the teleportation protocol, and suppose that the unknown qubit with Alice is entangled with another qubit in the possession of a third party, Charlie. Show how the protocol teleports the entanglement as well, i.e., at the end of the protocol, Bob's qubit is entangled with Charlie's.

9.3. Formulate the matrix equivalent of the dense coding protocol and show that it is unitary.

9.4. Analyzing the BB84 more thoroughly, consider that Eve measures every photon sent by Alice, in the $\mathbb{1}$ or H basis according to a random string m_e.

Suppose Alice and Bob now announce m bits out of their shared set. What is the probability that no error will be found? What fraction of Eve's bits would match with Alice's and Bob's? Would the public discussion between Alice and Bob help Eve at all?

9.5. At one point in history, it was suggested that Eve might benefit by measuring in a basis intermediate between the $\mathbb{1}$ and H:

$$|0_e\rangle = \cos\frac{\pi}{8}|0\rangle + \sin\frac{\pi}{8}|1\rangle \tag{9.10}$$

$$|1_e\rangle = \sin\frac{\pi}{8}|0\rangle - \cos\frac{\pi}{8}|1\rangle \tag{9.11}$$

What is the probability that any one measurement by Eve gives the correct result? If she prepares and then transmits photons in this basis to Bob, what is the probability that Bob's string has an error?

9.6. Suppose Alice prepares two qubits in the entangled state $\frac{1}{\sqrt{2}}\left[|01\rangle - |10\rangle\right]$ and sends one qubit to Bob. Suppose that Eve intercepts and measures that qubit, and then based on the outcome, prepares and sends a photon to Bob. What can you say about the correlation between the qubits with Alice and Bob?

Chapter 10

Quantum Error Correction

Bits as well as qubits are affected by noise in transmission, the process of computation or even storage. We will refer to the error-prone area as a *channel*. The error is introduced by unwanted but unavoidable interaction with the rest of the world and needs to be corrected before the data can be reliable. Classically this is important in communication systems and there is a whole thriving field of study of classical error correction. In quantum systems, states are intrinsically so fragile that unless they are impossibly isolated from the rest of the world, errors or noise would make computation or communication using them infeasible. Fortunately, it was discovered early in the history of this subject that it is possible to encode qubits in special ways so as to make the information resilient to errors. The qubits are decoded at the end of the computation/communication (schematic of Figure 10.1).

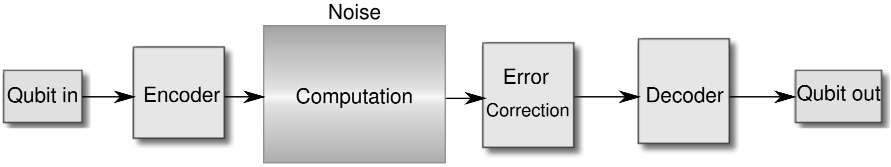


FIGURE 10.1: Simplified model of error correction.

Classically, the only errors that can occur in a binary computer are bit flips. The common way of protecting against these is to encode the information using redundant bits, that is, using multiple copies of the bits involved in the process. This way, corruption in the communication channel will affect some of the bits but in the end, we can measure the bits, analyze them for errors, and recover the encoded information by decoding. This is in a rough way analogous to repeating the message many times to ensure that the other party gets it right.

Now errors in a qubit are not just bit flips, which are merely one among an infinity of possible transformations. A qubit could accidentally undergo a change to any other point on the Bloch sphere. There is a whole continuum of possible changes that a qubit could go through. Encoding by repetition may seem inapplicable to qubits for two reasons. First, since measurement destroys quantum information, we cannot measure the states in the end to discover which error has occurred. Second, the no-cloning theorem prevents us from creating redundant quantum information by cloning a qubit. Also, one cannot think of copying an output before measuring it.

Nevertheless, as we will see in this section, it is possible to correct qubits for errors in an intrinsic manner without destroying the information they carry. It also turns out that the whole continuum of possible errors can be represented by a finite set of discrete errors. In short, quantum error correction is possible and efficiently implementable. We will study the basic principles of how this works in the somewhat artificial but simple context of single-qubit errors.

10.1 3-Qubit Repetition Code for Bit Flips

Classically, the repetition code is the simplest way of introducing redundancy to protect information. Assume that noise in the channel is modelled as a bit flip with probability p (and hence $1 - p$ for not flipping). This is schematised in Figure 10.2. This is known as the binary symmetric channel.

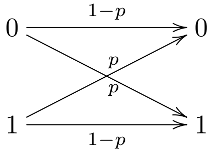


FIGURE 10.2: Binary symmetric channel for bit flips.

To protect against errors, each logical bit, indicated by the tilde, is encoded using three identical physical bits.

$$0 \to \tilde{0} = 000, \qquad 1 \to \tilde{1} = 111. \tag{10.1}$$

If p is sufficiently small, then the majority value decides what the original bit was. The total probability of error is the sum of probability that 2 bits flipped and that 3 bits flipped which is $3p^{(}1 - p) + p^3 = 3p^2 - 2p^3$. If $p < 1/2$, this is much smaller than the probability of error without encoding, which is p.

Now suppose we have a quantum channel that was susceptible to only qubit flips. We can model such a channel by X acting with probability p on a state passing through the channel. The quantum equivalent of the 3-bit repetition code represents each basis state by 3 identical qubits in the same basis state:

$$|0\rangle \to |\tilde{0}\rangle = |000\rangle; \qquad |1\rangle \to |\tilde{1}\rangle = |111\rangle \tag{10.2}$$

so that an arbitrary state is encoded as

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \longrightarrow |\tilde{\psi}\rangle = \alpha|000\rangle + \beta|111\rangle. \tag{10.3}$$

This encoding process is easily achieved by the circuit of Figure 10.3.

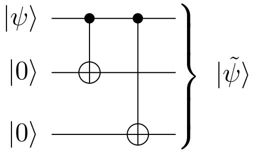


FIGURE 10.3: Encoding circuit for the 3-qubit bit-flip code.

This does not make three copies of the original state, however, and neither can we try to measure the state after it passes through the noise to check how it has changed, as that would destroy the superposition. If we assume that the channel is capable of flipping only one qubit, then the codeword state $|\tilde{\psi}\rangle$ could have changed into four possible output states:

$$S_0 : |\tilde{\psi}\rangle \to |\psi_0\rangle = \mathbb{1}|\tilde{\psi}\rangle = \alpha|000\rangle + \beta|111\rangle \tag{10.4a}$$
$$S_1 : |\tilde{\psi}\rangle \to |\psi_1\rangle = X_1|\tilde{\psi}\rangle = \alpha|001\rangle + \beta|110\rangle \tag{10.4b}$$
$$S_2 : |\tilde{\psi}\rangle \to |\psi_2\rangle = X_2|\tilde{\psi}\rangle = \alpha|010\rangle + \beta|101\rangle \tag{10.4c}$$
$$S_3 : |\tilde{\psi}\rangle \to |\psi_3\rangle = X_3|\tilde{\psi}\rangle = \alpha|100\rangle + \beta|011\rangle \tag{10.4d}$$

These states are known as *syndromes*, since we can diagnose the affliction due to noise by detecting which one occurred! But how do we detect the syndrome without measuring or copying? The way out is to use ancillary qubits, with controlled gates acting on them and to measure the ancillaries. We have to ensure that we do not get *any* information about the original state by this measurement, but still detect the syndrome. Note that the four syndrome states are all mutually orthogonal. Therefore it is possible to distinguish them by measuring a 2-qubit ancilla. Consider the circuit in Figure 10.4. Each qubit of the input state is indicated by its label.

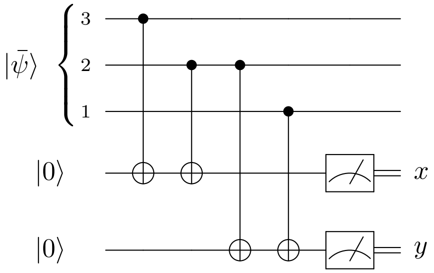


FIGURE 10.4: Syndrome measurement for the 3-qubit bit-flip code

TABLE 10.1: Syndrome measurement: outcomes.

Syndrome	xy
$\vert\psi_0\rangle$	00
$\vert\psi_1\rangle$	01
$\vert\psi_2\rangle$	11
$\vert\psi_3\rangle$	10

You can verify that the measured 2-bit number xy give you the syndrome as in Table 10.1.

The information contained in the input state $|\tilde{\psi}\rangle$ is not revealed by the measurements. It is easy to see that for each syndrome $|\psi_i\rangle$, the error can be corrected by applying X on the i^{th} qubit. This action can be linked to the xy values, which control the action of X on the corresponding qubit: $X^{x\bar{y}}$ on the first qubit, X^{xy} on the second, and $X^{\bar{x}y}$ on the last qubit of the codeword.

The nice thing about expressing it as this controlled action is that the process can be automated, bypassing the need for measurement, by applying suitable controlled gates as in Figure 10.5.

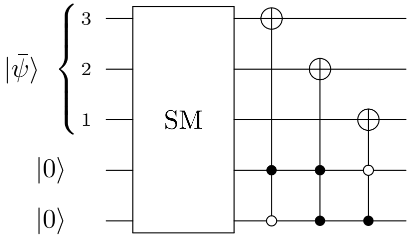


FIGURE 10.5: Error detection and correction for 3-qubit bit-flip code. Here SM is the syndrome measurement circuit.

10.1.1 Details: stabilizers

Why does this scheme work? It is possible to distinguish the syndromes, which are orthogonal states, if we measure a suitable observable of which they are eigenstates. It turns out that the bit-flip syndrome states $|\psi_i\rangle$ are eigenstates of the operators Z_1Z_2 and Z_2Z_3 with distinct sets of eigenvalues. In other words, for

$$\hat{O}_I = \mathbb{1} \otimes Z \otimes Z \text{ and } \hat{O}_{II} = Z \otimes Z \otimes \mathbb{1},$$

$$\hat{O}|\psi_i\rangle = \pm|\psi_i\rangle. \tag{10.5}$$

You can easily check that each $|\psi_i\rangle$ has a different set of eigenvalues for $\hat{O}_I$ and $\hat{O}_{II}$ (Table 10.2). These operators when acting on the full Hilbert space of

3-qubit states, do not change the subspaces containing the syndrome states. This subspace is said to be invariant under the action of these operators, which are therefore known as *stabilizers*. It is possible to understand why the

TABLE 10.2: Eigenvalues of stabilizers.

Error	Syndrome	Z_1Z_2	Z_2Z_3
$\mathbb{1}$	$\vert\psi_0\rangle$	$+1$	$+1$
X_1	$\vert\psi_1\rangle$	$+1$	-1
X_2	$\vert\psi_2\rangle$	-1	-1
X_3	$\vert\psi_3\rangle$	-1	$+1$

$\hat{O}$s of Equation 10.5 are the stabilizers and how they distinguish between the syndromes for single qubit-flip errors. The uncorrupted codeword state is unchanged by the action of $\hat{O}$. Each corrupted state is obtained by $|\psi_i\rangle = X_i|\tilde{\psi}\rangle$, and the operators Z_jZ_k either commute or anti-commute with X_i, depending on whether $i = j$ or k or not. It is a well-known concept in quantum mechanics that operators that commute or anti-commute with a transformation operator are symmetries of the system: the states are left unchanged by them. Therefore, the corrupted states can be distinguished by measuring $\hat{O}_I$ and $\hat{O}_I I$, without disturbing the states. Measuring Z_iZ_j is like comparing the values of the i^{th} and j^{th} qubits, giving +1 if they match and −1 if not. Recall that measuring a unitary operator $\hat{O}$ having eigenvalues ±1 is achieved by the circuit in Figure 10.6, with $|u\rangle$ an eigenstate of $\hat{O}$.

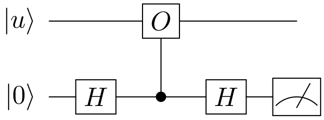


FIGURE 10.6: Circuit for measuring an operator $\hat{O}$.

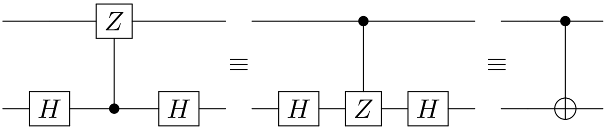


FIGURE 10.7: Circuit equivalences for measuring $\hat{Z}$.

If we need to measure Z_1Z_2, since the C-Z gate is symmetric, we can interchange the control and target qubits, and using $X = HZH$ (see Figure 10.7) and $H^2 = 1$, we can get the syndrome measurement circuit given by Figute 10.8. Check for yourself that each measurement in this process is identical to that in Figure 10.4.

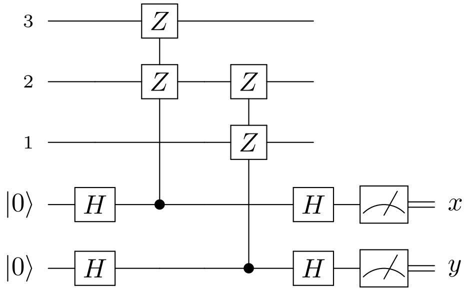


FIGURE 10.8: Measuring Z_1Z_2 and Z_2Z_3.

For a channel allowing a single qubit to change, we can estimate the minimum number of qubits needed to encode for error correction to work. We already noted that the corrupted states $|\psi_i\rangle$ are mutually orthogonal. Let's visualize these states in the Hilbert space of three qubits, which is 8-dimensional. Each of our corrupted states is a linear combination of two of the eight basis vectors of $\mathcal{H}^8$, lying in a 2-d plane. Each of these planes would be mutually orthogonal, since each of the component basis vectors is orthogonal to the other. This is the fact that underlies the success of the scheme described in the last section. If a codeword uses n qubits, there are $n+1$ syndromes including the uncorrupted state. Each syndrome is two-dimensional: we need $2(n+1)$ dimensions for the orthogonal subspaces of the 2^n-d Hilbert space in which the syndromes lie. Thus we need

$$2(n+1) \leq 2^n \implies n = 3 \text{ at least.} \tag{10.6}$$

Thus the 3-qubit encoding is the most basic possible scheme. Other schemes exist that utilize more qubits, and are more efficient, as we shall see.

10.1.2 Error analysis

Let us estimate the probability for the above technique to yield an uncorrupted state, considering a channel characterized by flipping of a qubit with probability $p < 1/2$. We list in Table 10.3 the probability of occurrence of various corrupted states in order of decreasing probability.

The probability that our procedure corrects errors is therefore

$$\mathcal{P}(\text{correct}) = (1-p)^3 + 3p(1-p)^2 = 1 - 3p^2 + 2p^3. \tag{10.7}$$

and the probability that we have an erroneous state is

$$\mathcal{P}(\text{incorrect}) = p^2(1-p) + p^3 = 3p^2 - 2p^3 < \mathcal{P}(\text{correct}). \tag{10.8}$$

TABLE 10.3: Probability of occurrence of corrupted states in a bit-flip channel.

Number of flips	states	probability
0	$\|\psi_0\rangle$	$(1-p)^3$
1	$X_1\|\psi_0\rangle$ $X_2\|\psi_0\rangle$ $X_3\|\psi_0\rangle$	$3p(1-p)^2$
2	$X_1X_2\|\psi_0\rangle$ $X_2X_3\|\psi_0\rangle$ $X_1X_3\|\psi_0\rangle$	$3p^2(1-p)$
3	$X_1X_2X_3\|\psi_0\rangle$	p^3

Box 10.1: Error Correction and Fidelity

People in the error-correcting business are not satisfied with this, and try to work out schemes that are better by comparing fidelities. We will see in Section 11.3.2 that the fidelity of two states is defined by their degree of overlap. If we start with a pure state $|\psi\rangle$, errors cause it to become a mixed state with probability p of transforming by X. This is represented by the density matrix

$$\rho_{\text{bf}} = pX|\psi\rangle\langle\psi|X + (1-p)|\psi\rangle\langle\psi|. \tag{10.9}$$

The fidelity of state transmission without error correction is given by

$$\begin{aligned} F &= \sqrt{\langle\psi|\rho_{\text{bf}}|\psi\rangle} && (10.10)\\ &= \sqrt{p\langle\psi|X|\psi\rangle^2 + (1-p)} && (10.11)\end{aligned}$$

This has a minimum value of $\sqrt{1-p}$, when the first term is zero. If we make use of the above protocol for error correction then for the 3-qubit encoded state,

$$\begin{aligned}\rho_{\text{corrected}} &= \left[(1-p)^3 + 3p(1-p)^2\right]|\psi\rangle\langle\psi| \\ &\quad + (\rho \text{ for 2 or more bit flips})\,, && (10.12)\end{aligned}$$

and the fidelity, using only the first two terms, is

$$F \geq \sqrt{(1-p)^3 + 3p(1-p)^2},$$

the same as the above.

10.2 Phase Flip Code

Bit flips alone are a very limited kind of error a qubit could undergo. Consider phase flips, which have no classical equivalent. A phase-flip quantum channel is defined as one that only allows single phase flips, that is, with probability p, $|1\rangle \to -|1\rangle$. Under the action of this channel, a generic state transforms as

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \to \alpha|0\rangle - \beta|1\rangle. \tag{10.13}$$

If we represent $|\psi\rangle$ in the X basis spanned by $|\pm\rangle = \frac{1}{\sqrt{2}}\left[|0\rangle \pm |1\rangle\right]$,

$$|\psi\rangle = \alpha'|+\rangle + \beta'|-\rangle, \tag{10.14}$$

then phase flip causes $|+\rangle \to |-\rangle$ and $|-\rangle \to |+\rangle$. Thus, the phase-flip case is unitarily equivalent to the bit-flip case since we can change basis to the X-basis by applying a H transform. Error correction can be followed just as in the bit-flip case, except that we now transform everything to the X basis by using the H gate at appropriate places. The 3-qubit encoding that will correct phase flip errors should then be

$$|0\rangle \to |+++\rangle, \qquad |1\rangle \to |---\rangle, \tag{10.15}$$

which is achieved by the circuit in Figure 10.9.

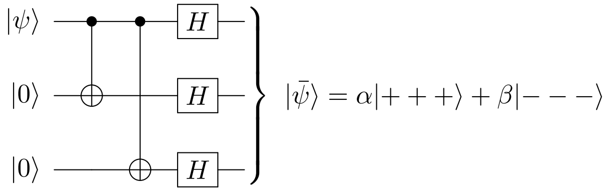


FIGURE 10.9: Encoding circuit for 3-qubit phase-flip code.

Syndrome measurement and recovery is now identical to the bit-flip case, except that we work in the X basis by applying an H gate to each qubit. The stabilizers are the operators

$$\hat{O}'_I = H^{\otimes 3} Z_1 Z_2 H^{\otimes 3} = X_1 X_2, \qquad \hat{O}'_{II} = H^{\otimes 3} Z_2 Z_3 H^{\otimes 3} = X_2 X_3. \tag{10.16}$$

Measuring these operators distinguishes the syndromes. This is like comparing the signs of the corresponding qubit values. Finally, recovery is performed by applying $HXH = Z$ to the appropriate qubit.

Exercise 10.1. Construct the circuit for detecting phase flip syndromes and for correcting them.

10.3 9-Qubit Shor Code

Let's now consider a channel that can produce both bit flips and phase flips. A code that combines bit and phase flip coding should protect against these errors. A simple way to do this is to first encode for phase flips:

$$|0\rangle \to |+++\rangle; \qquad |1\rangle \to |---\rangle$$

and then encode using the bit flip code:

$$|+\rangle \to \frac{1}{\sqrt{2}}\left(|000\rangle + |111\rangle\right); \qquad |-\rangle \to \frac{1}{\sqrt{2}}\left(|000\rangle - |111\rangle\right),$$

so that we have the final 9-qubit encoding

$$|0\rangle \to \frac{1}{2\sqrt{2}}\left(|000\rangle + |111\rangle\right)^{\otimes 3}; \quad |1\rangle \to \frac{1}{2\sqrt{2}}\left(|000\rangle - |111\rangle\right)^{\otimes 3}. \qquad (10.17)$$

Such a code is called a *concatenated* code, and this particular 9-qubit code was first proposed by Peter Shor. The circuit to achieve this encoding is obtained by *concatenating* the circuits for the phase flip and the bit flip encoding, as shown in Figure 10.10. The syndrome generators are easy to construct: bit-flips in each block can be detected by measuring $(Z_1Z_2, Z_2Z_3), (Z_4Z_5, Z_5Z_6)$ and (Z_7Z_8, Z_8Z_9). Further, phase flips between blocks can be distinguished by measuring $X_1X_2X_3X_4X_5X_6$ and $X_4X_5X_6X_7X_8X_9$.

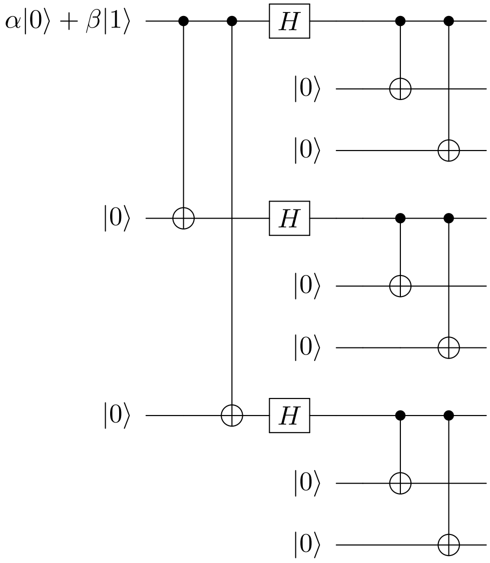


FIGURE 10.10: Encoding circuit for the 9-qubit Shor code

Exercise 10.2. Construct the circuit for error correction for this case.

Note that with eight stabilizers, we have a possibility of correcting for 2^8 different errors, but we have only tried to look at bit/phase flips of 9 qubits, which is $3 \times 9 + 1 = 28$! Thus this scheme is highly redundant. More efficient schemes using fewer encoding qubits have been proposed, and Shor's 9-qubit code is of purely historical interest now. The reason it is important to study this code is that it shows that it is possible to simultaneously correct for both bit flips and phase flips. Now it turns out that this will actually correct for *arbitrary* single-qubit errors, since, as we are about to show, any such error can be thought of as a combination of just bit flips and phase flips.

10.4 Discretization of Quantum Errors

One of the main results of the theory of quantum error correction is that any general quantum error can be composed only of discrete errors represented by the Pauli operators X, Z, and $Y = -iXZ$. Errors are induced on our qubit system due to effects of everything outside this system, which we will call the environment. The environment interacts weakly with the system to cause a change in the amplitudes of the basis states, a process called *decoherence* of the initial state. Initially, let's assume the system is created in a definite state $|\psi\rangle$. The environment has been excluded experimentally, so that the combined environment-qubit system is in a product state: $|e\rangle|\psi\rangle$. Subsequent interaction between the two results in a change of this state. In order to model this evolution, we represent the transformation of the computational basis states by

$$|e\rangle|0\rangle \longrightarrow |e_1\rangle|0\rangle + |e_2\rangle|1\rangle; \qquad (10.18a)$$

$$|e\rangle|1\rangle \longrightarrow |e_3\rangle|0\rangle + |e_4\rangle|1\rangle. \qquad (10.18b)$$

Here the kets $|e_i\rangle$ are (un-normalized) environment states that can be expressed as fractions of $|e\rangle$: $|e_i\rangle = a_i|e\rangle$. For example, the bit-flip error can be modelled with $a_1 = 0 = a_4, a_2 = 1 = a_3$ and the phase flip by $a_2 = 0 = a_3, a_1 = 1 = -a_4$. Now we want to be able to recognize the effect of such an evolution on the superposition state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, as an operation on the qubit system alone. In order to separate the effects on $|0\rangle$ and $|1\rangle$ we'll now write a general error in the terms of the projectors

$$\mathbb{P}_0 = |0\rangle\langle 0|, \quad \text{and} \quad \mathbb{P}_1 = |1\rangle\langle 1|. \qquad (10.19)$$

So we can write Equations 10.18 as

$$|e\rangle|0\rangle \longrightarrow \big(|e_1\rangle\mathbb{P}_0 + |e_2\rangle X\mathbb{P}_0\big)|0\rangle \qquad (10.20a)$$

$$|e\rangle|1\rangle \quad\longrightarrow\quad \big(|e_3\rangle X\mathbb{P}_1 + |e_4\rangle\mathbb{P}_1\big)|1\rangle. \tag{10.20b}$$

The error acting on $|\psi\rangle$ can be written as

$$|e\rangle|\psi\rangle \quad\longrightarrow \Big[\big(|e_1\rangle\mathbb{1} + |e_2\rangle X\big)\mathbb{P}_0 + \big(|e_3\rangle X + |e_4\rangle\mathbb{1}\big)\mathbb{P}_1\Big]|\psi\rangle. \tag{10.21}$$

Now the projection operators can be written in terms of the Pauli matrices:

$$\begin{aligned} Z = |0\rangle\langle 0| - |1\rangle\langle 1|; &\qquad \mathbb{1} = |0\rangle\langle 0| + |1\rangle\langle 1|; \\ \implies \mathbb{P}_0 = \frac{\mathbb{1}+Z}{2}; &\qquad \mathbb{P}_1 = \frac{\mathbb{1}-Z}{2}. \end{aligned} \tag{10.22}$$

Also, using $XZ = iY$, we get

$$|e\rangle|\psi\rangle \longrightarrow \big(|E_1\rangle\mathbb{1} + |E_2\rangle\hat{X} + |E_3\rangle\hat{Y} + |E_4\rangle\hat{Z}\big)|\psi\rangle, \tag{10.23}$$

where we have appropriately regrouped the environment states $|e_i\rangle$ to obtain the new environment states $|E_i\rangle$. (We do not care about the exact form of these states since we are not going to observe them.) We thus see that the generic error can be expressed as a linear combination of the discrete errors corresponding to the action of the Pauli matrices.

If we encode using n qubits for error-correction, then a generic state would become

$$|e\rangle|\tilde{\psi}\rangle_n \longrightarrow \left(|d\rangle\mathbb{1} + \sum_{i=1}^{n}(|a_i\rangle\hat{X}_i + |b_i\rangle\hat{Y}_i + |c_i\rangle\hat{Z}_i)\right)|\tilde{\psi}\rangle_n. \tag{10.24}$$

In order to diagnose the syndromes, the 2^n-d Hilbert space must admit at least $1+3n$ 2-d subspaces:

$$2^{n-1} \geq 1 + 3n,$$

so $n = 5, 7, 9...$

Thus the minimum codeword size is 5 qubits. We can see now that the 9-qubit Shor code is not efficient; we can make do with fewer qubits.

10.5 The 5-Qubit Code

The number of syndromes in a 5-qubit scheme would be $5 \times 3 + 1 = 16$. We'd thus need 4 stabilizer operators since $2^4 = 16$. We'll simply give the operators (see Mermin [48] or Laflamme et al. [44]):

$$\begin{aligned} M_0 = Z_1X_2X_3Z_4, &\qquad M_2 = Z_3X_4X_0Z_1, \\ M_1 = Z_2X_3X_4Z_0, &\qquad M_3 = Z_4X_0X_1Z_2. \end{aligned} \tag{10.25}$$

These operators satisfy

$$M_0 M_1 M_2 M_3 = \mathbb{1}. \tag{10.26}$$

We can see that each operator flips 2 qubits, and the encoding is more usefully defined in terms of these:

$$\begin{aligned} |\bar{0}\rangle &= \frac{1}{4}(\mathbb{1}+M_0)(\mathbb{1}+M_1)(\mathbb{1}+M_2)(\mathbb{1}+M_3)|00000\rangle, && (10.27a) \\ |\bar{1}\rangle &= \frac{1}{4}(\mathbb{1}+M_0)(\mathbb{1}+M_1)(\mathbb{1}+M_2)(\mathbb{1}+M_3)|11111\rangle. && (10.27b) \end{aligned}$$

One thing to notice is that $|\bar{0}\rangle$ is composed of 16 basis states, each with an even number of 1's, while $|\bar{1}\rangle$ is composed of states with an even number of 0's, so that the states are mutually orthogonal. Each M_i commutes or anti-commutes with the X_i, Y_i, and Z_i error operators, so that the fifteen syndromes and the uncorrupted state are distinguished by different sets of ±1 eigenvalues of the M's. Measuring them would therefore diagnose the syndromes.

Exercise 10.3. Compute the 5-qubit codewords.

Exercise 10.4. Verify that the circuit of Figure 10.11 performs the 5-qubit encoding.

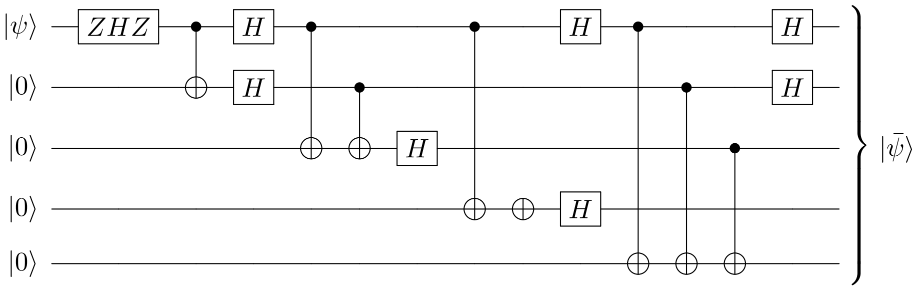


FIGURE 10.11: The encoding circuit for the 5-qubit code

As you are probably feeling, this code is harder to analyze and less transparent than the Shor code. For practical purposes, the 7-qubit code due to Steane is more popular.

10.6 The 7-Qubit Code

We again give the stabilizers, codewords for the logical bit states and the encoding circuit, for completeness. You can refer to the text by Mermin [48] for a full discussion on how the scheme works to correct errors. The 7-qubit code is stabilized by 6 operators that distinguish the syndromes due to X, Y, or Z acting on any one qubit. These are the Steane operators:

$$\begin{aligned} N_0 &= X_0X_3X_5X_6; & N_3 &= Z_0Z_3Z_5Z_6; \\ N_1 &= X_1X_3X_5X_6; & N_4 &= Z_1Z_3Z_5Z_6; \\ N_2 &= X_2X_3X_5X_6; & N_5 &= Z_2Z_3Z_5Z_6. \end{aligned} \tag{10.28}$$

Observe that they mutually commute, and $N_i^2 = \mathbb{1}$. The 7-qubit encoding is defined by the operations

$$|\bar{0}\rangle = \frac{1}{\sqrt{8}}(\mathbb{1} + N_0)(\mathbb{1} + N_1)(\mathbb{1} + N_2)|0\rangle_7 \tag{10.29a}$$

$$|\bar{1}\rangle = \frac{1}{\sqrt{8}}(\mathbb{1} + N_0)(\mathbb{1} + N_1)(\mathbb{1} + N_2)|1\rangle_7. \tag{10.29b}$$

You can see that $|\bar{0}\rangle$ is a state with an odd number of 0's while $|\bar{1}\rangle$ has an even number. The usefulness of this code lies in the easy way in which many 1-qubit operations generalize to operations on the 7-qubit codewords. For instance, defining

$$\bar{X} = X^{\otimes 7}, \quad \bar{Z} = Z^{\otimes 7}, \quad \bar{H} = H^{\otimes 7}, \tag{10.30}$$

we find that

$$\bar{X}|\bar{0}\rangle = |\bar{1}\rangle; \qquad \bar{Z}|\bar{0}\rangle = |\bar{0}\rangle; \qquad \bar{H}|\bar{0}\rangle = \frac{1}{\sqrt{2}}(|\bar{0}\rangle + |\bar{1}\rangle); \tag{10.31a}$$

$$\bar{X}|\bar{1}\rangle = |\bar{0}\rangle; \qquad \bar{Z}|\bar{1}\rangle = -|\bar{1}\rangle; \qquad \bar{H}|\bar{1}\rangle = \frac{1}{\sqrt{2}}(|\bar{0}\rangle - |\bar{1}\rangle). \tag{10.31b}$$

This makes it a lot more convenient to use this encoding in various circuits.

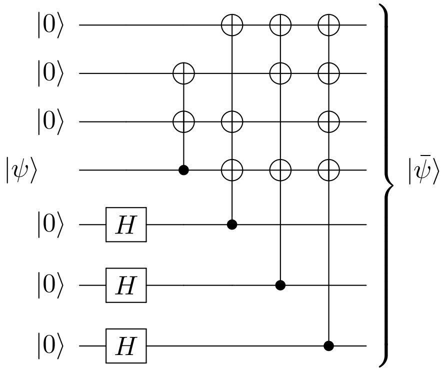


FIGURE 10.12: Circuit for the 7-qubit encoding. The qubits are arranged according to significance from highest to lowest, top to bottom.

You can also show that the rather cute circuit of Figure 10.12, again due to Mermin [48], performs this encoding.

Exercise 10.5. Draw a circuit to measure the syndromes for the 7-qubit code.

While we have discussed the basic reasons for the success of quantum error-correcting codes, we have barely scratched the surface of this complex and intriguing field. In general, errors need not be restricted to single-qubit errors. Nor need they be unitary. The full theory of quantum error-correcting codes is beyond the scope of this book. That theory examines how the system can be embedded in a larger system with a number of entangled qubits. Measurement of some of the ancilla qubits can lead to error diagnosis and correction. A very readable account of this is given in the book by Reiffel and Polak [58].

Another important subject we are not dealing with is *fault tolerant computation.* The assumption in all we have studied so far is that the gates and circuits we employ are potentially error-free in themselves. This can hardly be guaranteed in practice. However by special coding a circuit or a gate can be made tolerant to errors.

Problems

10.1. Suppose that a channel introduces a linear combination of bit-flip errors on a qubit encoded by the 3-qubit repetition code. This is modelled as $\alpha\hat{X}_1 + \beta\hat{X}_2$. Show that the syndrome measurement for the 3-bit code correctly distinguishes the possible syndromes in this case as well.

10.2. Construct the matrix representation for the syndrome detection operation (before the measurement of the ancillas).

10.3. Construct the unitary operator for error recovery using the 3-qubit phase flip code.

10.4. Construct the table of eigenvalues of the stabilizers for the 9-qubit Shor code, analogous to Table (10.2).

Chapter 11

Characterization of Quantum Information

When we manipulate physical systems for various purposes, we are essentially encoding and decoding the information content in those systems in precisely defined terms tailored to our purpose. Conversationally speaking of information, one thinks of the "new knowledge" gained when a particular physical process is completed, such as watching television, reading an article, or measuring the output voltage at the end of a circuit. When we get used to the idea that information is not something abstract that is a result of cognition, but is actually physically carried by the system that's being observed or measured, we are closer to a scientific understanding of information.

Classically, information theory gained respectability when Claude Shannon [63] quantified the information content in a physical system or communication channel. He was working at Bell Labs at that time and was interested in optimizing telephonic communication. He also realized the importance of information in the context of data compression and cryptography. We will start with his theory, and see how it can be adapted to describing the information content of a quantum system. For a more in-depth treatment of the subject, you can refer to the excellent book by Barnett [4]. Matters regarding quantum information are beautifully discussed in the work by Mark Wilde [73], and in Chapter 5 of Preskill's lecture notes [57].

11.1 Measures of Information

We would like to develop a measure for the rather abstract concept of information, so that efficiencies of different protocols or physical systems of communication can be compared, and more efficient systems designed.

We need to first have a model for the process we are describing, and Shannon's proposed model has stuck (Figure 11.1). We start with a **source** of information, which, like a talking person or a buzzing telephone receiver, generates **messages** using some predefined language consisting of symbols we will call the **alphabet**. The alphabet could, for example, be the set of English

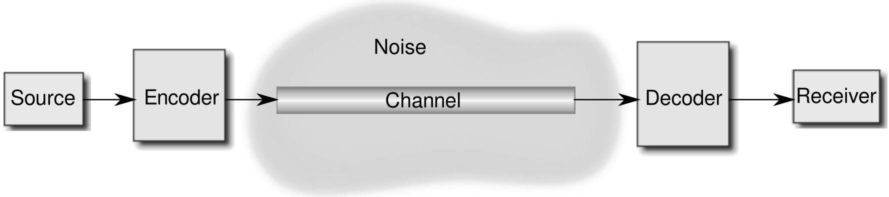


FIGURE 11.1: Simplified model for communication

letters for communicating through a written message, or a set of “dit”s and “da”s for a message in Morse code, or 0’s and 1’s for a computerized message.

Any message emitted by the source is then **encoded** for transmission using the physical system involved. For a talking person, the message is encoded in the vibration of the air molecules. For a telephonic message, the encoding is in terms of analog electric pulses in the wire. For the now-obsolete telegraph, encoding was done in binary dit’s and da’s represented as short or long electric pulses which we have now refined into the 0s and 1s of the modern binary computer. The encoded message is then transmitted across a **channel**. This is obvious as the air carrying the spoken message, the wire carrying the telegraph signal, or the optical cable transmitting long distance digital messages. In the context of quantum information, the channel may well just be the environment that the quantum system finds itself in between computational steps.

The importance of the channel in information theory lies in how much it costs in terms of its usage, and how it may distort or reduce the quality of the message being sent. We would need to quantify the compression of the message in order to more efficiently utilize the resources, as well as the capacity of the channel to carry information in the presence of **noise**. Noise could be literal in the case of a talking person, random electrical signals in a telephone or telegraph wire or the computer circuitry, or the change in the state of a quantum system interacting inadvertently with its environment. At the other end of the communication model is the **decoder** and finally the receiver of the message, which is either the ear of the audience, the ear-piece of the telephone, the printed output of the telegraph, or the monitor of the computer.

We will now see how we can develop a measure for the information carried by a message, or a symbol in the message. The appearance of a symbol at the receiver’s end is an event, whose probability can be predicted if we have some idea of the properties of the source. We will talk in the language of events and their probabilities of occurrence.

11.1.1 Classical picture: Shannon entropy

According to Shannon, information associated with events is related to their probability of occurrence. Let's take a simple example. Suppose a magician has hidden a ball in one of three boxes labeled 1, 2, and 3. Now he asks you to choose the box in which is ball is hidden. How would you choose? It depends on the information you have about the ball's location. In the beginning, you do not know which box it is in, so you think it is equally probable to be in any of the three boxes. Thus each box carries 1/3 of the information about which box the ball is in. You'd alternatively say that the ball is in each box with equal *probability.* The situation is thus described by an initial *probability distribution* $\mathcal{P}_{in}$:

$$\mathcal{P}_{in}(\text{ball is in 1}) = 1/3, \quad \mathcal{P}_{in}(\text{ball is in 2}) = 1/3, \quad \mathcal{P}_{in}(\text{ball is in 3}) = 1/3.$$

An event such as opening any one box now will give you further information on where the ball is among the three. Suppose you open one box, say box 2, and the ball is not in it. The probability distribution has now changed: conditioned by the event of having opened box 2:

$$\mathcal{P}_2(\text{ball is in 1}) = 1/2, \quad \mathcal{P}_2(\text{ball is in 2}) = 0, \quad \mathcal{P}_2(\text{ball is in 3}) = 1/2.$$

Opening a box now gives information only about where the ball is in the two.

On the other hand, what about the information with the magician for the same situation? He knows that he has put the ball in box1, so his distribution is

$$\mathcal{P}_m(\text{ball is in 1}) = 1, \quad \mathcal{P}_m(\text{ball is in 2}) = 0, \quad \mathcal{P}_m(\text{ball is in 3}) = 0.$$

In this case opening a box adds no information at all to the magician's knowledge!

This example teaches us a few things:

The information carried by an event is related inversely to its probability of occurrence before it has occurred. (After the event, the probability is of course 1!) The more probable the occurrence, the less information the event carries. The occurrence of an event removes doubts about the possibilities before it occurs. The information it carries is thus the doubt it removes by occurring. The information carried by an event is changed if a previous event carries related information. If an event has happened, it carries no *new* information as compared to an event that has not yet happened.

Let's now try to quantify the information $\mathcal{I}$ carried by an event $\mathcal{E}$. If we are talking about messages encoded in symbols sent across a channel, then the event is the reading of the symbol by the receiver. The occurrence of a particular symbol x is the event $\mathcal{E}(x)$ with probability $p(x)$, $\sum_x p(x) = 1$. Now the mathematical formulation of information is concerned with the syntactic form of the message rather than the semantic. This means that we are not going to worry about the *meaning* conveyed by a message in a literal sense (which would depend on how the read symbol is translated by the receiver's brain), but rather by the form of the message itself, in terms of the symbols it carries. In other words, information carried by a symbol does not depend on which symbol it is, but only on our ignorance, or uncertainty, about its occurrence, before it is read. Thus the information carried by the symbol x on the occurrence of event $\mathcal{E}(x)$, must depend inversely on $p(x)$.

There are some properties we expect the information function to have. Suppose two events $\mathcal{E}_1$ and $\mathcal{E}_2$ both occur. The information carried by this joint occurrence is $\mathcal{I}(\mathcal{E}_1 + \mathcal{E}_2)$. If the result of one event $\mathcal{E}_1$ is revealed, then the information carried by the other event must be the difference: $\mathcal{I}(\mathcal{E}_2) = \mathcal{I}(\mathcal{E}_1 + \mathcal{E}_2) - \mathcal{I}(\mathcal{E}_1)$. Thus, the information carried by the joint event is the sum of the information carried by each. So if many events occur sequentially, then the information also builds up in the same order.

Remember that if an event is certain to occur, then it carries no information. A certainty is represented by probability 1, so that $\mathcal{I}(1) = 0$. (That's like a computer that is switched off, so it reveals no information!)

Collecting all these properties together, we require our mathematical information $\mathcal{I}(\mathcal{E}_i)$ to be a function that satisfies

1. inverse relation to probability: $\mathcal{I}(\mathcal{E}_i) \sim \dfrac{1}{p_i}$,
2. monotonously increasing, continuous function of p_i,
3. additivity: $\mathcal{I}(\mathcal{E}_1 + \mathcal{E}_2) = \mathcal{I}(\mathcal{E}_1) + \mathcal{I}(\mathcal{E}_2)$,
4. identity corresponding to information of certainty, $\mathcal{I}(1) = 0$.

One function that satisfies all these properties is the logarithm. This led Shannon to define the **self-information** of an event $\mathcal{E}_i$ as

$$\mathcal{I}(\mathcal{E}_i) = \frac{1}{\log p_i} = -\log p_i. \tag{11.1}$$

Here, the base of the logarithm denotes the units in which information is measured. If we use the natural logarithm then the unit of information is *nats*. If all events are binary in nature (yes-no answers), then the logarithm is taken to the base 2 and the units of information is *bits*. Note that the difference between different units for information is a multiplicative constant since

$$\log_b x = \log_b a \log_a x.$$

Example 11.1.1. Let's calculate the information in bits carried by the drawing of a card out of a playing deck of 64 cards. To rephrase the problem, suppose a magician asks you to pull out a card at random, and then tries to guess which card you drew. The logical way for the magician to remove his ignorance about the card would be to ask you questions about the card. (Of course he cannot ask you which card it is!) Since we want the answer in bits, the answers to these questions must be binary: "yes" or "no." The problem then translates to how many such binary-answer questions the magician must ask for correctly guessing the number. The procedure he adopts is the "binary search" algorithm of dividing the range of possible answers into two at each step and asking if the card is in one of the two ranges. Your yes or no will allow him to select one of the sections and further divide it. Consider the following sample scenario:

Q 1. Is it between 1 and 32? Ans: No.

Q 2. ... between 33 and 49? Ans: Yes.

Q 3. ... 33 and 41? Ans: No.

⋮

Q n.

What is the total number n of such questions? It's the number of times the range [1 – 64] can be bifurcated, which, if you follow through the above sequence of questions, will be 6.

$$n = 6 = \log_2 64 = -\log_2 \frac{1}{64},$$

and 1/64 is the probability of choosing one particular card out of the deck. Clearly this answer is independent of which card it is (the semantic meaning of the event).

A **message** is the occurrence of a string of events: the appearance of each symbol constituting the message. Thus the total information carried by a message is the weighted average of all the symbols in the message. This is given by

$$H(m) = \sum_i p_i \mathcal{I}(\mathcal{E}_i) = -\sum_i p_i \log p_i. \tag{11.2}$$

What if a particular symbol doesn't occur in the message? In that case too, $p = 0$. Even though $\log p$ is then undefined, the symbol cannot contribute to the information carried by the message, and for our purposes, we define $\mathbf{0}\ \log \mathbf{0} \equiv \mathbf{0}$.

From its similarity to the thermodynamic quantity of the same name, the information function $H(m)$ has been called the **entropy** of the message. In statistical thermodynamics, we seek to relate the macroscopic properties system to the *microstates* of its constituents. Notably, the energy of a gas is related to the momenta of its constituent molecules. If the number of microstates compatible with a given macrostate is Ω then the Boltzmann entropy of the system is defined to be

$$S = k_B \ln \Omega,$$

where k_B is Boltzmann's proportionality constant.[1] We think of the microstates as tiny imaginary cells dividing the total gas volume, for the gas particles to occupy. If p_i is the probability that the i^{th} cell is occupied, the Boltzmann entropy of the gas can be worked out to be

$$S = -k_b \sum_i p_i \ln p_i. \tag{11.3}$$

Example 11.1.2. Consider a set X of possible events $\{a, b, c, d\}$ with the following probabilities:

$$p(a) = 0.5;\ p(b) = 0.3;\ p(c) = 0.1;\ p(d) = 0.1$$

Then the entropy of this distribution is

$$H(X) = -0.5 \log(0.5) - 0.3 \log(0.3) - 2 \times 0.1 \log(0.1) = 1.685 \text{ bits}.$$

Exercise 11.1. What is the information carried by the toss of an unbiased coin?

Exercise 11.2. How does the above change if the coin is biased?

11.1.2 Mathematical characteristics of the entropy function

We will denote by an *ensemble* X the collection of events (represented by a random variable) x occurring with probability $p(x)$:

$$X \equiv \{x, p(x)\}. \tag{11.4}$$

Definition 11.1. *The entropy function for an ensemble X is given by*

$$H(X) = -k \sum_x p(x) \log p(x). \tag{11.5}$$

The number k is a constant which depends on the units in which H is measured.

The function $H(X)$ satisfies the following properties.

1. $H(X)$ is always positive, and is continuous as a function of $p(x)$ that is symmetric under exchange of any two events x_i and x_j.

[1] See footnote 1 on page 111.

2. It has a minimum value of 0, when only one event occurs with probability 1 and all the rest have probability 0. This is obvious to see since $H(p)$ is a positive function and its minimum has to be zero.

3. It has a maximum value of $k \log n$ when each x occurs with equal probability $1/n$. Here is a simple proof of this fact:

$$\begin{aligned} H(X) - k\log n &= k\sum_x p(x)\log\frac{1}{p(x)} - k\sum_x p(x)\log n \\ &= k\sum p(x)\log\frac{1}{np(x)} \\ &\leq k\sum p(x)\left(\frac{1}{np(x)} - 1\right). \end{aligned}$$

This is because $\log x \leq x - 1$, with equality only if $x = 1$ (see Figure 11.2), an important result often used in information theoretic proofs.

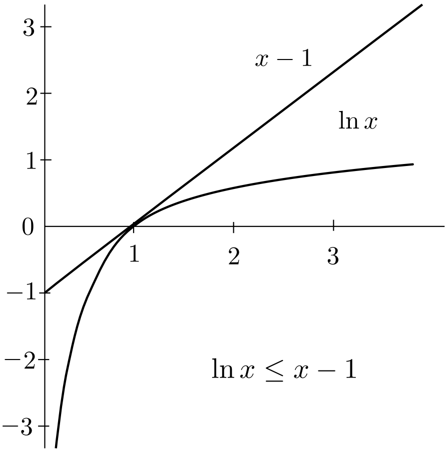


FIGURE 11.2: Graph of $y = x - 1$ compared with $y = \ln x$.

So we have

$$\begin{aligned} H(X) - k\log n &\leq k\left(\sum\frac{1}{n} - \sum p(x)\right) = 0, \\ \therefore H(X) &\leq k\log n. \end{aligned} \tag{11.6}$$

Box 11.1: Binary Entropy

A very useful concept is the entropy function of a probability distribution of a binary random variable, such as the result of the toss of a coin, not necessarily unbiased. Here, one value occurs with probability p and the other with $1-p$. We then have

$$H_{\text{bin}}(p) \quad = \quad -p\log p - (1-p)\log(1-p). \tag{11.7}$$

In this simple case all the listed properties of the mathematical entropy function are obvious.

1. Positive: $H_{\text{bin}}(p) > 0$ always;
2. Symmetric: $H_{\text{bin}}(p) = H_{\text{bin}}(1-p)$;
3. $H_{\text{bin}}(p)$ has a maximum of 1 when $p = 1/2$ as in a fair coin;
4. $H_{\text{bin}}(p)$ has a minimum of 0 when $p = 1$ as in a two-headed coin.

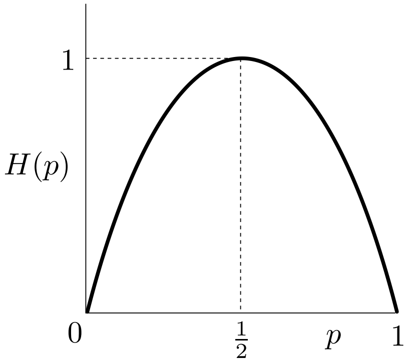


FIGURE 11.3: The binary entropy function.

This function is a useful tool in deriving properties of entropy, especially when different probability distributions are mixed together. An important property of the entropy function is made evident in this simple case: that of **concavity**. This property is used very often in concluding various results in classical as well as quantum information theory. The graph in Figure 11.3 shows that the function is literally concave. A mathematical statement of this property is that the function lies above any line cutting the graph. Algebraically, for two points $x_1, x_2 < 1$, we have

$$H_{\text{bin}}\big(px_1 + (1-p)x_2\big) \quad \geq \quad pH_{\text{bin}}(x_1) + (1-p)H_{\text{bin}}(x_2). \tag{11.8}$$

11.1.3 Relations between entropies of two sets of events

From the way it is defined, Shannon entropy is closely related to probability theory. In this book, we do not expect a thorough background in probability theory, so I will simply draw your attention to some important results, so that you may be piqued enough to look them up on your own. Consider two ensembles $X = \{x, p(x)\}$ and $Y = \{y, p(y)\}$. We will define various measures to compare the probability distributions $\{p(x)\}$ and $\{p(y)\}$.

1. **Relative entropy** of X and Y measures the difference between the two probability distributions $\{p(x)\}$ and $\{p(y)\}$:

$$\begin{aligned} H(X \parallel Y) &= -\sum_{x,y} p(x) \log p(y) - H(X) \\ &= \sum_{x,y} p(x) \log \frac{p(y)}{p(x)}. \end{aligned} \tag{11.9}$$

   Here again we use the convention that

$$-0 \log 0 \equiv 0, \quad -p(x) \log 0 \equiv \infty, p(x) > 0. \tag{11.10}$$

   An important property of the relative entropy is that it is positive. The relative entropy is also called the *Kullback–Leibler distance.* However, it is not symmetric, and so is not a true distance measure, but it gives us, for example, the error in assuming that a certain random variable has probability distribution $\{p(y)\}$ when the true distribution is $\{p(x)\}$. Thus this definition is more useful when we have a set of events X with two different probability distributions $\{p(x)\}$ and $\{q(x)\}$,

$$H(p \parallel q) = \sum_{x} p(x) \log \frac{p(x)}{q(x)}. \tag{11.11}$$

2. **Joint entropy** of X and Y measures the combined information presented by both distributions. Classically, the joint probability of X and Y, denoted by $\{p(x, y)\}$, is defined over a set $X \otimes Y$. The joint entropy is then

$$H(X, Y) = -\sum_{x,y} p(x, y) \log p(x, y). \tag{11.12}$$

   If X and Y are independent events, then

$$H(X, Y) = H(X) + H(Y). \tag{11.13}$$

3. **Conditional entropy** measures the information gained by the occurrence of X if Y has already occurred and we know the outcome. The

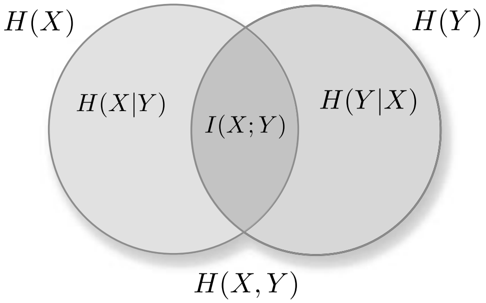


FIGURE 11.4: Relationship between entropic quantities.

classical conditional probability of an event x given y is defined as $p(x|y) = p(x,y)/p(y)$, and we have

$$\begin{aligned} H(X|Y) &= -\sum_{x,y} p(x|y) \log p(x|y) && (11.14) \\ &= H(X,Y) - H(Y). && (11.15) \end{aligned}$$

The second equation is an important relation: a chain rule for entropies:

$$H(X,Y) = H(X) + H(Y|X). \tag{11.16}$$

4. **Mutual information** measures the correlation between the distributions of X and Y. This is the difference between the information gained by the occurrence of X, and the information gained by occurrence of X if Y has already occurred. The mutual information is symmetric, so we have

$$\begin{aligned} I(X;Y) &= H(X) - H(X|Y) = H(Y) - H(Y|X) && (11.17) \\ &= H(X) + H(Y) - H(X,Y). && (11.18) \end{aligned}$$

The mutual information is a measure of how much the uncertainty about X is reduced by a knowledge of Y. You can also see that it is the relative entropy of the joint distribution $p(x,y)$ and the product distribution $p(x)p(y)$:

$$I(X;Y) = \sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)}. \tag{11.19}$$

One way to picture the interrelationships between these entropic quantities is the Venn diagram of Figure 11.4. Given these definitions, the Shannon entropies satisfy the following properties that are easily proved:

1. Relative entropy is non-negative:

$$H\left(p(x) \parallel q(x)\right) \geq 0, \tag{11.20}$$

with equality iff $p(x) = q(x)$.

2. Mutual information is non-negative:

$$I(X;Y) \geq 0 \tag{11.21}$$

with equality iff X and Y are independent.

3. Conditioning reduces entropy:

$$H(X|Y) \leq H(X), \tag{11.22}$$

with equality iff X and Y are independent.

4. Subadditivity:

$$H(X,Y) \leq H(X) + H(Y), \tag{11.23}$$

with equality iff X and Y are independent.

Example 11.1.3. The ancient Indian game of dice used two cuboids marked 1 to 4 on the long faces. When the pair is rolled, the results are two independent sets of events; let's call them X and Y. Suppose a trickster uses loaded dice with the following joint probabilities $p(x,y)$:

$p(x,y)$	1	2	3	4	$p(x)$
1	$1/16$	$1/16$	$1/8$	$1/4$	$1/2$
2	$1/32$	$1/32$	$1/16$	$1/8$	$1/4$
3	$1/32$	$1/32$	$1/32$	$1/32$	$1/8$
4	$1/8$	0	0	0	$1/8$
$p(y)$	$1/4$	$1/8$	$7/32$	$13/32$	1

The individual probability distributions for X and Y are known as **marginals** and are calculated as the sum of probabilities for one variable over all the values of the other. The marginals for X and Y are indicated in the last row and column of the table above.

1. Entropies of the two marginal distributions: $\sum p_i \log_2 1/p_i$ for each marginal.

$$H(X) = 1.75 \text{ bits}, \quad H(Y) = 1.88 \text{ bits}.$$

2. Joint entropy: $\sum p_i \log_2 1/p_i$ for all the 16 entries in the table.

$$\begin{aligned} H(X,Y) &= \frac{1}{4}\log_2 4 + 3\left(\frac{1}{8}\log_2 8\right) + 3\left(\frac{1}{16}\log_2 16\right) + 6\left(\frac{1}{32}\log_2 32\right) \\ &= 3.31 \text{ bits.} \end{aligned}$$

Note that this is less than $H(X) + H(Y)$. The difference is the mutual information: $I(X;Y) = 0.32$.

3. Conditional entropy: for $H(X|Y)$ you need the entropies of the conditional distributions, $H(X|Y = y)$, $p(x|y) = \frac{p(x,y)}{p(y)}$.

$$\begin{aligned} H(X|Y) &= \sum_y p(y)H(X|Y = y) \\ &= \frac{1}{4}H\left(\frac{1}{4}, \frac{1}{8}, \frac{1}{8}, \frac{1}{2}\right) + \frac{1}{8}H\left(\frac{1}{2}, \frac{1}{4}, \frac{1}{4}, 0\right) \\ &\quad + \frac{7}{32}H\left(\frac{4}{7}, \frac{2}{7}, \frac{1}{7}, 0\right) + \frac{13}{32}H\left(\frac{8}{13}, \frac{4}{13}, \frac{1}{13}, 0\right) \\ &= 0.4375 + 0.1875 + 0.3016 + 0.5033 = 1.43 \text{ bits,} \\ &= H(X,Y) - H(Y). \end{aligned}$$

Similarly, $H(Y|X) = 1.56$ bits $= H(X,Y) - H(X)$.

4. Mutual information

$$I(X;Y) = H(Y) - H(Y|X) = 0.32 \text{ bits} = H(X) - H(X|Y).$$

Once these ideas were introduced, Shannon went ahead to define the limits on compression of a given message source, and the capacity of a channel. We will state these results for completeness, but will not prove them.

To optimize the use of channel resources, we often encode the messages from a source so as to reduce their average lengths. This process is therefore called compression. If we assume that the transmission is completely faithful, that is, the message is not distorted in transmission, then we say that the channel is noiseless. In this situation, how small can we make the encoded message without losing the original information it carries?

Theorem 11.1. ***Shannon's noiseless coding theorem****: The maximum compression that can be achieved for a given source is given by its entropy.*

This theorem applies to the average length of a message from a source S having entropy $H(S)$:

$$L_{\text{av}} \geq H(S). \tag{11.24}$$

It should be understood in context: the actual entropy of the source of messages is seldom known: one only has access to the messages it produces. If we could use some means to guess at the entropy of the source, then that number is the maximum information the source is capable of, and we cannot compress below it, for doing so would entail losing some information. In transmitting a message across a channel, the entropy of the message is often also called the *entropy rate.*

Shannon then considers the effect of noise on a channel: it can change the information content of the message. He defines the **channel capacity** C of a noisy channel as follows. Suppose Alice and Bob are communicating over this channel and the messages Alice sends are denoted by the ensemble X. The messages Y received by Bob are not necessarily the same as X since the channel is noisy. The capacity of the channel is then the maximum mutual information of X and Y:

$$C = \max H(X : Y). \tag{11.25}$$

Theorem 11.2. ***Shannon's channel coding theorem:***

A channel with capacity C can be used to transmit (messages in some appropriate coding) at any rate $R < C$ with error that can be reduced to an arbitrarily small amount.

If the rate is larger than C then the error cannot be reduced below some limit.

These two theorems complement each other in the field of communications. We try to optimize the use of resources, and while encoding to entropic limit minimizes the size of a message, noisy channels force redundancy for error correction and the second theorem provides a limit for that.

There is an excellent treatment of all these ideas in the book by Cover and Thomas [21], especially the second chapter.

11.2 The von Neumann Entropy

The principles of entropy as applied to information were actually originally formulated by John von-Neumann, in his famous mathematical book on the foundations of quantum mechanics [70] (first published in German in 1932). It was he who suggested that Shannon name his information measure as "entropy." The von Neumann entropy is a measure of the information carried by quantum systems, and is expected to be related to the maximum compression possible while encoding information in quantum states. If we are using pure orthogonal quantum states to encode information, the entropy of the system is the same as Shannon's entropy. The main difference from classical information arises due to the impossibility of reliably distinguishing non-orthogonal states. Thus if we have a mixed state ρ for representing information, the von Neumann entropy $S(\rho)$ is required to quantify the information content. As we will see, the important concept of entanglement is also quantified by the von Neumann entropy.

For a quantum state, the probabilities for measurement results are given by the density operator described in Chapter 5. When ρ is expressed in the orthonormal basis $\{|i\rangle\}$ which diagonalizes it,

$$\rho = \sum_i \lambda_i |i\rangle\langle i|, \tag{11.26}$$

λ_i represent the probability of the system being found in the state $|i\rangle$. The entropy of the system is therefore

$$S(\rho) = -\sum_i \lambda_i \log \lambda_i, \tag{11.27}$$

which also can be written as

$$S(\rho) = -\operatorname{Tr}(\rho \log \rho). \tag{11.28}$$

Here we have used the definition of Equation 3.15 for the logarithm of a matrix. Since the density matrix is positive, its logarithm always exists. This is von Neumann's definition of the entropy of a quantum system. Note that the base of the logarithm in this case is the dimension of the Hilbert space of the constituent systems. So we naturally use the log base 2 for qubits.

Example 11.2.1. Consider a mixed state consisting of $|+\rangle$ with probability 1/4 and $|-\rangle$ with probability 3/4. The density matrix is

$$\begin{aligned} \rho &= \frac{1}{4}|+\rangle\langle+| + \frac{3}{4}|-\rangle\langle-| \\ &= \frac{1}{8}\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} + \frac{3}{8}\begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \\ &= \frac{1}{4}\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}. \end{aligned}$$

The eigenvalues of this matrix are $\frac{1}{4}$ and $\frac{3}{4}$, so that the von Neumann entropy is

$$S(\rho) = \frac{1}{4}\log_2 4 + \frac{3}{4}\log_2 \frac{4}{3} = 0.81 \text{ qubits.}$$

11.2.1 Properties of the von Neumann entropy

Some properties of the von Neumann entropy immediately follow from the definition.

1. The minimum value of $S(\rho)$, zero, occurs for pure states.

$$S(\rho) \quad \geq \quad 0. \tag{11.29}$$

Thus even though a pure state embodies probabilities of measurement outcomes, the information carried by it is zero since it represents a definite vector in Hilbert space.

2. The maximum value of $S(\rho)$ is $\log d$, where d is the dimensionality of the Hilbert space.

$$S(\rho) \quad \leq \quad \log d. \tag{11.30}$$

This occurs for maximally mixed states with each ρ_i taking the value $1/d$. You will prove this in an exercise.

3. Invariance under unitary transformations:

Under unitary evolution U of the quantum system, the von Neumann entropy remains unchanged.

$$S(U\rho U^\dagger) = S(\rho). \tag{11.31}$$

4. Entropy of preparation:

We can think of entropy as a measure of mixedness of the system, or its departure from purity. When constructing a state ρ out of an ensemble of pure states $|x\rangle$ with probability $p(x)$, in general we will find that

$$H(X) \geq S(\rho). \tag{11.32}$$

That is, the Shannon (classical) entropy is greater than the von Neumann entropy. The equality (Equation 11.27) holds when the $|x\rangle$ are mutually orthogonal. The interpretation of this result is that when viewed in a basis in which ρ is not diagonal, we are not in the same basis in which the system was prepared. Measurement results in such a basis will have probabilities such that the entropy is more than the von Neumann entropy. The latter is therefore called the *entropy of preparation* of the system.

Example 11.2.2. For a state that is 25% $|0\rangle$ and 75% $|+\rangle$, the Shannon entropy is

$$H(X) = \frac{1}{4}\log 4 + \frac{3}{4}\log\frac{4}{3} = 0.81 \text{ bits.}$$

The density matrix is

$$\rho \quad = \quad \frac{1}{4}\begin{pmatrix}1 & 0\\ 0 & 0\end{pmatrix} + \frac{3}{8}\begin{pmatrix}1 & 1\\ 1 & 1\end{pmatrix} = \frac{1}{8}\begin{pmatrix}5 & 3\\ 3 & 3\end{pmatrix}$$

with eigenvalues $1/2 \pm 1/4\sqrt{5/2}$, so that the von Neumann entropy is

$$S(\rho) = 0.485 \text{ qubits.}$$

5. Entropy and measurement: When an observable $\hat{A}$ is measured in a state ρ, the outcomes a have a probability distribution

$$\mathcal{P}(a) = \langle a|\rho|a\rangle, \tag{11.33}$$

where $|a\rangle$ is the eigenstate of $\hat{A}$. We then find that the Shannon entropy of the measurement outcomes is greater than the von Neumann entropy of the state:

$$H(A) \geq S(\rho), \tag{11.34}$$

with equality when $\hat{A}$ commutes with ρ. This means that measurement increases the randomness in the system unless we measure a commuting observable.

11.2.2 Entropy of composite systems

Some of the properties of the von Neumann entropy for composite systems are similar to those of Shannon entropy, while some others are quite different. We discuss a few here.

1. Concavity: $S(\rho)$ is a concave function. That is, for a linear combination of states $\rho = c_1\rho^A + c_2\rho^B$, the resulting entropy is usually greater than the weighted sum of the individual entropies:

$$S(\rho) \geq c_1 S\left(\rho^A\right) + c_2 S\left(\rho^B\right). \tag{11.35}$$

The physical interpretation is that as when two systems are mixed, the resultant is more uniform than each of the individual systems. To prove this, you need to remember that the logarithm is not a linear function. It is, in fact, a concave function (look at the graph of Figure 11.2). This also means that the function $x\log x$ is concave. In the basis $\{|i\rangle\}$ in which ρ is diagonal, $\rho_i = \langle i|\rho|i\rangle$. Let's introduce the notation $\rho_i^A = \langle i|\rho^A|i\rangle$ etc.

Proof.

$$\begin{aligned}
\rho_i \log \rho_i \quad &\geq \quad c_1\rho_i^A \log \rho_i^A + c_2\rho_i^B \log \rho_i^B \\
\implies S(\rho) \quad &= \quad -\sum_i \rho_i \log \rho_i \\
&\geq \quad -\sum_i (c_1\rho_i^A \log \rho_i^A + c_2\rho_i^B \log \rho_i^B) \\
&\quad\;\; = c_1 S(\rho^A) + c_2 S(\rho^B).
\end{aligned}$$

□

2. Quantum relative entropy. Suppose that $\{|i\rangle\}$ and $\{|m\rangle\}$ are two sets of orthogonal bases for the Hilbert space of the system. For density operators

$$\rho = \sum_i p_i|i\rangle\langle i|; \qquad \sigma = \sum_m q_m|m\rangle\langle m|,$$

we can define the relative entropy as

$$S(\rho \parallel \sigma) = \mathrm{Tr}\big[\rho(\log\rho - \log\sigma)\big]. \tag{11.36}$$

In evaluating this quantity, we find that it is always non-negative: a result sometimes known as Klein's inequality.

Proof.

$$\begin{aligned}
S(\rho \parallel \sigma) &= \sum_i \langle i|\rho(\log\rho - \log\sigma)|i\rangle \\
&= \sum_i p_i \log p_i - p_i\langle i|\log\sigma|i\rangle. && (11.37) \\
\text{Here, } \langle i|\log\sigma|i\rangle &= \langle i|\sum_m \log q_m |m\rangle\langle m|i\rangle \\
&= \sum_m \log q_m P_{im} && (11.38) \\
\text{where} \quad & P_{im} \equiv \langle i|m\rangle\langle m|i\rangle && (11.39) \\
&\geq 0; \qquad \sum_i P_{im} = 1 = \sum_m P_{im} && (11.40) \\
& \text{(Such a matrix is called } \textit{doubly stochastic.}) \\
\text{So, } S(\rho \parallel \sigma) &= \sum_i p_i \left[\log p_i - \sum_m P_{im} \log q_m\right] && (11.41) \\
&= \sum_{i,m} p_i P_{im} \log \frac{p_i}{q_m} \ \text{(since} \sum_m P_{im} = 1) \\
&\geq \sum_{i,m} p_i P_{im} \left(1 - \frac{q_m}{p_i}\right) \ \text{(since } \log x \geq 1 - \frac{1}{x}) \\
& = 0, \ \text{(using Eq 11.41)} \\
\Longrightarrow S(\rho \parallel \sigma) &\geq 0. && (11.42)
\end{aligned}$$

□

3. Subadditivity. Given two systems A and B with joint state ρ^{AB}, and reduced density matrices ρ^A and ρ^B, the **joint entropy** defined simply as

$$S(\rho^{AB}) \equiv -\mathrm{Tr}\rho^{AB} \log \rho^{AB} \tag{11.43}$$

satisfies

$$S(\rho^{AB}) \leq S(\rho^A) + S(\rho^B), \tag{11.44}$$

with equality only when the two systems are uncorrelated. Thus entanglement reduces the entropy, i.e., increases the information, of the system.

Proof. The proof follows as an application of Klein's inequality for $\rho = \rho^{AB}$ and $\sigma = \rho^A \otimes \rho^B$. Suppose $|i\rangle$ and $|m\rangle$ are bases for the Hilbert spaces of A and B, respectively. From Klein's inequality,

$$\begin{aligned} S(\rho^{AB}) \quad &\leq \quad -\mathrm{Tr}\rho^{AB} \log(\rho^A \otimes \rho^B) \\ &\quad = -\mathrm{Tr}\rho^{AB} \log \rho^A - \mathrm{Tr}\rho^{AB} \log \rho^B \end{aligned}$$

Now the first term in this is

$$-\langle i, m|\rho^{AB} \log \rho^A |i, m\rangle \quad = \quad -\mathrm{Tr}_A \rho^A \log \rho^A = S(\rho^A).$$

Similarly for the other term. So we have

$$S(\rho^{AB}) \quad \leq \quad S(\rho^A) + S(\rho^B) \tag{11.45}$$

□

There is another result, the triangle inequality also known as the Araki–Lieb inequality, that can be similarly proved:

$$S(\rho^{AB}) \geq |S(\rho^A) - S(\rho^B)|. \tag{11.46}$$

4. Conditional entropy.

$$S(A|B) \quad \equiv \quad S(\rho^{AB}) - S(\rho^B). \tag{11.47}$$

While Shannon conditional entropy can never be negative, the von Neumann entropy can, for systems that are entangled [16]. This can be proved to be a criterion for entanglement.

There are many more inequalities and properties of the von Neumann entropy that can be proved, for which we refer you to Nielsen and Chuang [50], the book by Ohya and Petz [51] and the review article by Wehrl [71].

11.3 Distance Measures

An important consideration in information theory is the comparison of two systems: probability distributions in the classical context and states (pure or mixed) in the quantum. For such comparisons, various measures collectively labeled *distance* measures have been proposed. We'll consider some of them here, to educate ourselves in the concepts involved.

11.3.1 Kolmogorov or trace distance

A sort of distance between two probability distributions $p(x)$ and $q(x)$ for the same random variable X can be defined as

$$D(p(x), q(x)) = \frac{1}{2}\sum_x |p(x) - q(x)|. \tag{11.48}$$

This is similar to a "metric" for determining the distance between points in a space.

One context in which such a measure is useful is in a dynamic process, where information X is sent through a (noisy) channel and appears as Y. We wish to compute the probability of error in the channel by comparing the two distributions. To do this, we first make a copy of the input and call it X', and then look at the probability distribution of the pairs (X', X) and (X', Y). Let's compute the trace distance between these two distributions $p(x) = p(X' = x, X = x)$ and $q_y = p(X' = x, Y = y)$:

$$\begin{aligned} D(p,q) &= \frac{1}{2}\sum_{x,y} |p(x) - q(y)| \\ &= \frac{1}{2}\sum_{x \neq y} p(x) + \frac{1}{2}\sum_x |p(x) - q(x)| \\ &= \frac{1}{2}\left(p(X' \neq Y) + 1 - p(X' = Y)\right) \\ &= p(X \neq Y) \end{aligned}$$

For two quantum states ρ and σ, we can define the Kolmogorov distance using the trace function

$$D(\rho, \sigma) = \frac{1}{2}\,\mathrm{Tr}|\rho - \sigma|. \tag{11.49}$$

How do we compute this? We will define the mod of a matrix A by

$$|A| = \sqrt{A^2} = \sqrt{A^\dagger A},$$

where the last equality holds if A is Hermitian, which is true of density matrices. We can easily see how this reduces to the classical distance, if we can diagonalize ρ and σ in the same basis to write

$$\rho = \sum_x p(x)|x\rangle\langle x|, \quad \sigma = \sum_x q(x)|x\rangle\langle x|.$$

Two matrices can be simultaneously diagonalized if and only if they commute. Then we see that

$$D(\rho, \sigma) = \frac{1}{2}\,\mathrm{Tr}\left|\sum_x (p(x) - q(x))|x\rangle\langle x|\right|$$

$$\begin{aligned} &= \frac{1}{2}\sum_x |p(x)-q(x)| \ \mathrm{Tr}|(|x\rangle\langle x|)| \\ &= \frac{1}{2}\sum_x |p(x)-q(x)| \\ &= D(p(x),q(x)). \end{aligned}$$

Example 11.3.1. It may be instructive to visualize the trace distance between single qubits by a Bloch sphere picture. Let our states be represented by Bloch vectors $\vec{\boldsymbol{p}}$ and $\vec{\boldsymbol{q}}$:

$$\rho = \frac{1}{2}\left(\mathbb{1} + \vec{\boldsymbol{p}}\cdot\vec{\boldsymbol{\sigma}}\right), \quad \sigma = \frac{1}{2}\left(\mathbb{1} + \vec{\boldsymbol{q}}\cdot\vec{\boldsymbol{\sigma}}\right).$$

The trace distance is then

$$D(\rho,\sigma) \quad = \quad \frac{1}{4}\ \mathrm{Tr}|(\vec{\boldsymbol{p}} - \vec{\boldsymbol{q}})\cdot\vec{\boldsymbol{\sigma}}|$$

The matrix $(\vec{\boldsymbol{p}} - \vec{\boldsymbol{q}})\cdot\vec{\boldsymbol{\sigma}} = \vec{\boldsymbol{a}}.\vec{\boldsymbol{\sigma}}$ has eigenvalues $\pm a$. So the eigenvalues of $|\vec{\boldsymbol{a}}.\vec{\boldsymbol{\sigma}}|$ are $|a|$ and $\mathrm{Tr}|\vec{\boldsymbol{\sigma}}| = 2|a|$. So we have

$$D(\rho,\sigma) = \frac{1}{2}\left(|\vec{\boldsymbol{p}} - \vec{\boldsymbol{q}}|\right)$$

which is half of the geometric distance between the points $\vec{\boldsymbol{p}}$ and $\vec{\boldsymbol{q}}$ in the Bloch ball.

The trace distance can be interpreted as follows: if two quantum states are close in trace distance, then when measurements are performed in those states, the resulting probability distributions are close in the classical trace distance.

11.3.2 Fidelity

Another important measure for comparing probability distributions is the **fidelity**, which is easily extended to quantum states. This is variously defined in different texts, but we will stick to a simple operational definition here:

$$\mathcal{F}(p(x),q(x)) = \sum_x \sqrt{p(x)q(x)}. \tag{11.50}$$

The square root is used so that we have $\mathcal{F}(p(x),p(x)) = 1$. This definition is compatible with the inner product of two vectors with components $\{p(x)\}$ and $\{q(x)\}$.

In the quantum case, the fidelity between a pure state $|\psi\rangle$ and a state $|\phi\rangle$ is the inner product:

$$\mathcal{F}(\psi, \phi) = \langle\phi|\psi\rangle \tag{11.51}$$

$|\mathcal{F}|^2$ can also be thought of as the probability of confusing the state $|\psi\rangle$ with $|\phi\rangle$ in an experimental situation. Another way of looking at it is that if the state $|\psi\rangle$ is sent through a communication protocol, the probability that the end state $|\phi\rangle$ is the same as the input state is (the mod-square of) the fidelity of the process. The fidelity is minimum, 0, if the two states are orthogonal, and maximum, 1, if the two states are identical. Classically, these are the only two situations that could possibly arise. But in the quantum world, there exists a continuity of states connecting the two possibilities, and this distinguishes quantum information from classical.

One can extend this definition to mixed states as well: for states ρ and σ,

$$\mathcal{F}(\rho, \sigma) \quad = \quad \mathrm{Tr}(\sqrt{\rho\sigma}). \tag{11.52}$$

Example 11.3.2. If we have a pure state $|\psi\rangle$ and a mixed state ρ, we can calculate the fidelity as

$$\begin{aligned}
\mathcal{F}(|\psi\rangle, \rho) &= \mathrm{Tr}(\sqrt{|\psi\rangle\langle\psi|\rho}) \\
&= \mathrm{Tr}(\sqrt{\langle\psi|\rho|\psi\rangle}) \\
&= \sqrt{\langle\psi|\rho|\psi\rangle} \qquad (11.53)
\end{aligned}$$

Example 11.3.3. If two density matrices ρ and σ commute then they can be diagonalized in the same basis and the fidelity can be calculated as

$$\begin{aligned}
\mathcal{F}(\rho, \sigma) &= \mathrm{Tr}\sqrt{\sum_x (p(x)q(x))|x\rangle\langle x|} \\
&= \mathrm{Tr}\sum_x \sqrt{p(x)q(x)}|x\rangle\langle x| \\
&= \sum_x \sqrt{p(x)q(x)}|x\rangle\langle x| = \mathcal{F}(p(x), q(x)). \qquad (11.54)
\end{aligned}$$

Fidelity is not a distance, but can be used to define one between density operators, the so-called Bures distance

$$\mathcal{D}_B = \sqrt{2 - 2\mathcal{F}}, \tag{11.55}$$

which is a metric on the space of states.

11.4 Entanglement Measures

Owing to the importance of entanglement as a resource in quantum information processing, it is necessary to construct measures of entanglement between two component systems. We saw in Chapter 4 a condition for the separability of 2-qubit states. For a generic higher dimensional density matrix to be **separable**, a test known as the *positive partial transpose* (PPT) condition was proposed by Peres [55] and the Horodecki's [42]. The density matrix of the system can be expressed as

$$\rho^{AB} = \sum_{i,j,l,m} p_{ijlm} |i\rangle\langle j| \otimes |l\rangle\langle m|. \tag{11.56}$$

where $|i\rangle, |j\rangle$ are basis states for system A, while $|l\rangle, |m\rangle$ are those of B. The *partial transpose* with respect to system B is obtained by interchanging the row and column indices of the second system:

$$\rho^{\mathrm{T}_B} \equiv \sum_{i,j,l,m} p_{ijlm} |i\rangle\langle j| \otimes |m\rangle\langle l|. \tag{11.57}$$

For separable states, this operator is *positive*, i.e., has non-negative eigenvalues only. If this operator has a negative eigenvalue then the state represented by ρ^{AB} is entangled.

Example 11.4.1. It is easy to see that the partial transpose of a separable density operator has no negative eigenvalue:

$$\rho^{AB} = \sum_i p_i \rho_i^A \otimes \rho_i^B, \tag{11.58}$$

Taking partial transpose with respect to B is just taking the transpose of the reduced matrix ρ_i^B. This action does not alter the eigenvalues of ρ_B and hence those of ρ^{AB}, which were non-negative to start with.

Exercise 11.3. Show that the partial transposes of the density matrices for the Bell states have a negative eigenvalue.

Entanglement has so far only been described qualitatively, and we know of the two extremes of separable states and maximally entangled 2-qubit states. We'd like to develop measures for entanglement that are more quantitative and generic. We expect any entanglement measure $E(\rho)$ to have the following properties.

1. For an unentangled state, $E(\rho) = 0$.

2. *Local* unitary transformations on the system should leave the entanglement unchanged.

3. If non-unitary operations are included (for example measurement), then the entanglement cannot increase.

Many different entanglement measures have been proposed, useful in different contexts.

1. Distance measures between the given state and the "nearest" unentangled state can be directly used.

2. Entropy of entanglement: If the system at hand (A) is considered as a component of a pure state ρ^{AB}, expressed in Schmidt form,

$$\rho^{AB} \quad = \quad \sum_i \lambda_i |i^A\rangle\langle i^A| \otimes |i^B\rangle\langle i^B|. \tag{11.59}$$

the entropy of the reduced density matrix for A is a measure of its entanglement with B:

$$E(A) \quad = \quad S(\,\mathrm{Tr}_B \rho^{AB}) = -\sum_i |\lambda_i|^2 \log|\lambda_i|^2, \tag{11.60}$$

The entropy for the reduced density matrix of B is also the same. Clearly, if the two states were unentangled, then they will be pure states themselves and the entropy would be zero. This measure also satisfies the other two conditions above. Thus, an entanglement measure for a pure composite state is the von Neumann entropy of any of the reduced density matrices.

This measure is, however, not applicable for mixed states, since the von Neumann entropy of a subsystem can be non-zero even if the states are not entangled.

3. Entanglement of formation: Since entanglement is created when the system are prepared, one common measure of entanglement is the entanglement of formation of the entangled pair. Suppose one is to prepare an ensemble of states in a given entangled state ρ. In one interpretation, the entanglement of formation measures the number of Bell states required to construct this state. If ρ is constructed out of a mixture of pure states $\{\phi_i\}$, we have

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

Each state $|\psi_i\rangle$ has its own entropy of entanglement E_i. This decomposition is not unique, and we have to choose the *minimum* out of all possible decompositions to define the entropy of formation of ρ:

$$\mathcal{E}(\rho) = min\left[\sum_i p_i E_i(|\psi_i\rangle)\right]. \tag{11.61}$$

4. Concurrence: This is a somewhat less intuitive measure of entanglement but is widely used and is related to the entanglement of formation discussed above. It was first proposed by Wootters in 1998 [75].

We saw in Chapter 4 that a 2-qubit pure state

$$|\psi\rangle \quad = \quad \alpha|00\rangle + \beta|01\rangle + \gamma|10\rangle + \delta|11\rangle, \tag{11.62}$$

is separable only if $\alpha\delta = \beta\gamma$ (Equation 4.9). The difference $|\alpha\delta - \beta\gamma|$ can be taken to be a measure of entanglement. One way to obtain this is to consider

$$\begin{aligned} |\tilde{\psi}\rangle &= Y_A \otimes Y_B |\psi^*\rangle, & (11.63)\\ C(\psi) &= |\langle\psi|\tilde{\psi}\rangle| & (11.64)\\ &= 2|\alpha\delta - \beta\gamma| & (11.65) \end{aligned}$$

This can be extended for a mixed state with density matrix ρ^{AB}: define

$$\tilde{\rho} = \hat{Y}_A \otimes \hat{Y}_B \rho^* \hat{Y}_A \otimes \hat{Y}_B,$$

then concurrence can be defined as

$$C(\rho) = max(0, \lambda_1 - \lambda_2 - \lambda_3 - \lambda_4), \tag{11.66}$$

where the λ_i are the square roots of the eigenvalues of $\rho\tilde{\rho}$ in decreasing order. For two-qubit systems, it turns out that the entanglement of formation is related to the concurrence:

$$E(\rho) = h\left(\frac{1}{2}\left(1 + \sqrt{1 - C^2}\right)\right), \tag{11.67}$$

where $h(x)$ is the standard entropy of a binary probability distribution:

$$h(x) = -x\log(x) - (1-x)\log(1-x).$$

These measures have dealt only with bipartite entanglement: entanglement between two subsystems. There are many more ideas dealing with entanglement of mixed states that are not discussed here. Neither is the much more complex scenario of multipartite entanglement.

Problems

11.1. What is the information carried by a throw of a die with 6 faces? What is the information carried by n throws of the same die?

11.2. An experiment produces photons with a 60% probability of being right circularly polarized and 40% of being left circularly polarized. Find the entropy (i) in an experiment to test for circular polarization; (ii) in an experiment to test for linear polarization.

11.3. Derive the mutual information relation of Equation 11.18 if the definition is Equation 11.19.

11.4. Consider a preparation of photons that has 70% probability of producing right circular polarization and 30% probability of producing vertical polarization.

(a) Construct the density matrix for the prepared photon state and find its eigenvalues.

(b) What is the physical meaning of the eigenvectors of this matrix?

(c) Find the entropy of this system.

11.5. Prove that for pure states, $\rho^2 = \rho \implies S(\rho) = 0$.

11.6. Prove the Araki–Lieb inequality, Equation 11.46.

11.7. Prove using the Klein inequality that for a d dimensional system, $S(\rho) \leq \log d$.

11.8. Calculate the concurrence for the Bell state $|\beta_{11}\rangle$.

Bibliography

[1] Yakir Aharonov and Daniel Rohrlich. *Quantum Paradoxes – Quantum Theory for the Perplexed.* Wiley-VCH, 2005.

[2] Alain Aspect, P. Grangier, and G. Roger. Experimental tests of realistic local theories via Bell's theorem. *Physical Review Letters*, 47:481, 1981.

[3] Adriano Barenco, Charles H. Bennett, Richard Cleve, David P. DiVincenzo, Norman Margolus, Peter Shor, Tycho Sleator, John Smolin, and Harald Weinfurter. Elementary gates for quantum computation. *Physical Review A 52*, page 3457, 1995.

[4] Stephen M. Barnett. *Quantum Information.* Oxford Master Series in Atomic, Optical amd Laser Physics. Oxford University Press, 2009.

[5] John S. Bell. On the Einstein–Podolsky–Rosen paradox. *Physics*, 3:195 – 200, 1966.

[6] C.H. Bennett, G. Brassard, C. Crepeau, R. Josza, A. Peres, and W.K. Wootters. Teleporting an unknown quantum state via dual classical and Einstein–Podolsky–Rosen channels. *Physical Review Letters*, 70:1895, 1993.

[7] Charles H. Bennett. Notes on the history of reversible computation. *IBM Journal of Research and Development*, 32(1):1623, 1988.

[8] Charles H. Bennett, Françios Bessette, Gilles Brassard, Louis Salvail, and John Smolin. Experimental quantum cryptography. *Journal of Cryptology*, 5:3–28, 1992.

[9] Charles H. Bennett, Gilles Brassard, and Artur K. Ekert. Quantum cryptography. *Scientific American*, October:50–57, 1992.

[10] Charles H. Bennett and Stephen J. Wiesner. Communication via one- and two-particle operators on Einstein–Podolsky–Rosen states. *Physical Review Letters*, 69(20):2881–2884, November 1992.

[11] Ethan Bernstein and Umesh Vazirani. Quantum complexity theory. *Proceedings of the 25th Annual ACM Symposium on the Theory of Computing*, page 1120, 1993.

[12] Antoine Berut, Artak Arakelyan, Artyom Petrosyan, Sergio Ciliberto, Raoul Dillenschneider, and Eric Lutz. Experimental verification of Landauer's principle linking information and thermodynamics. *Nature*, 483:187–189, 2012.

[13] Arno Bohm. *Quantum Mechanics: Foundations and Applications.* Springer Verlag, 3 edition, 2001.

[14] Niels Bohr. Discussion with Einstein on epistemological problems in atomic physics. In *Quantum Theory and Measurement*, pages 9–49. Princeton University Press, 1983.

[15] Gilles Brassard, Peter Hoyer, and Alain Tapp. Quantum counting. *arXiv:quant-ph/9805082*, 1998.

[16] Nicolas Cerf and Chris Adami. Negative entropy and information in quantum mechanics. *Physical Review Letters*, 79, 1997.

[17] J. F. Clauser, M. A. Horne, A. Shimony, and R. A. Holt. Proposed experiment to test local hidden-variable theories. *Physical Review Letters*, 23 (15):8804, 1969.

[18] J. F. Clauser and A. Shimony. Bell's theorem: experimental tests and implications. *Reports on Progress in Physics*, 41:1881, 1978.

[19] Richard Cleve, Artur Ekert, Chiara Machiavello, and Michele Mosca. Quantum algorithms revisited. *Proceedings of the Royal Society of London A*, 454:339, 1998.

[20] Claude Cohen-Tannoudji, Bernard Diu, and Frank Laloe. *Quantum Mechanics, Vol 1 and Vol 2.* Wiley and Sons, 1977.

[21] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory.* Wiley, 1991.

[22] D David Bohm. A suggested interpretation of the quantum theory in terms of hidden variables, part i and ii. *Physical Review*, (2) 85:166179, 180–193, 1952.

[23] C. M. Dawson and M. A. Nielsen. The Solovay–Kitaev algorithm. *arxiv:quant-ph/0505030*, 2005.

[24] David Deutsch. Quantum theory, the Church–Turing principle and the universal quantum computer. *Proceedings of the Royal Society of London, A*, 400, 1985.

[25] David Deutsch. Quantum computational networks. *Proceedings of the Royal Soceity of London, A*, 425, 73, 1989.

[26] David Deutsch and Richard Josza. Rapid solutions of problems by quantum computation. *Proceedings of the Royal Society of London A*, 439, 1992.

[27] Dennis Dieks. Communication by EPR devices. *Physics Letters A*, 92 (6):271272, 1982.

[28] W. Diffie and M. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22 (6):644–654, 1976.

[29] Paul A. M. Dirac. *Quantum Mechanics.* OUP, 1958.

[30] David P. DiVincenzo. The physical implementation of quantum computation. *Fortschritte der Physik*, 48:771, 2000.

[31] Albert Einstein, Boris Podolsky, and Nathan Rosen. Can quantum-mechanical description of physical reality be considered complete? *Physical Review*, 47:777, 1935.

[32] Artur Ekert. Quantum cryptography based on Bell's theorem. *Physical Review Letters*, 67:661 – 663, 1991.

[33] Richard P. Feynman. *The Feynman Lectures on Computation.* Addison-Wesley, 1996.

[34] Richard P. Feynman, R. B. Leighton, and M. Sands. *The Feynman Lectures on Physics, Vol III.* Addison-Wesley, 1965.

[35] Edward Fredkin and Tommaso Toffoli. Conservative logic. *International Journal of Theoretical Physics*, 21 (3-4), 1982.

[36] Walther Gerlach and Otto Stern. Das magnetische moment des silberatoms. *Zeitschrift fur Physik*, 9:353, 1922.

[37] G. C. Ghirardi, A. Rimini, and T. Weber. Unified dynamics for microscopic and macroscopic systems. *Physical Review D*, 34:470, 1986.

[38] Nicolas Gisin, Gregoire Ribordy, Wolfgang Tittel, and Hugo Zbinden. Quantum cryptography. *Review of Modern Physics*, 74:145–195, 2002.

[39] Daniel Greenberger, Michael Horne, and Anton Zeilinger. Going beyond Bell's theorem. In *Bell's Theorem, Quantum Theory, and Conceptions of the Universe.* Kluwer, Dordrecht, 1989. available on arxiv: arXiv:0712.0921 [quant-ph].

[40] L. K. Grover. Quantum mechanics helps in searching for a needle in a haystack. *Physical Review Letters*, 79, no. 2:325 328, 1997.

[41] Douglas Hofstadter. *Gödel, Escher, Bach: an Eternal Golden Braid.* Penguin, 1979.

[42] M. Horodecki, P. Horodecki, and R. Horodecki. Separability of mixed states: Necessary and sufficient conditions. *Physics Letters A*, 223:1, 1996.

[43] Phillip Kaye, Raymond Laflamme, and Michele Mosca. *An Introduction to Quantum Computing.* Oxford University Press, 2007.

[44] E. Knill, R. Laflamme, R. Martinez, and C. Negrevergne. Benchmarking quantum computers: the five-qubit error correcting code. *Physical Review Letters*, 86:5811–4, 2001.

[45] Lev Landau. On the problem of damping in wave mechanics. *Zeitschrift fur Physik*, 35:430, 1927.

[46] Rolf Landauer. Irreversibility and heat generation in the computing process. *IBM Journal of Research and Development*, 5:183–191, 1961.

[47] Ralph C. Merkle. Secure communications over insecure channels. *Communications of the ACM*, 21 (4):294–299, 1978.

[48] N. David Mermin. *Quantum Computer Science: An Introduction.* Cambridge University Press, 2007.

[49] Eugen Merzbacher. *Quantum Mechanics.* John Wiley, 3rd edition, 1998.

[50] Michael A. Neilsen and Isaac L. Chuang. *Quantum Computation and Quantum Information.* Cambridge University Press, 2002.

[51] Masanori Ohya and Denes Petz. *Quantum Entropy and Its Uses.* Springer-Verlag, 1993.

[52] Arun K. Pati and Samuel L. Braunstein. Impossibility of deleting an unknown quantum state. *Nature*, 404:164, 2000.

[53] Roger Penrose. *The Emperor's New Mind.* Oxford, 1989.

[54] Asher Peres. *Quantum Theory: Concepts and Methods.* Kluwer Academic Press, 1977.

[55] Asher Peres. Separability criterion for density matrices. *Physical Review Letters*, 77:14131415, 1996.

[56] T.E. Phipps and J.B. Taylor. The magnetic moment of hydrogen atoms. *The Physical Review*, 29 (2):309–320, 1927.

[57] John Preskill. *Lecture notes for Physics 219: Quantum Computation.* http://www.theory.caltech.edu/~preskill/ph229/, 2004.

[58] Eleanor G. Rieffel and Wolfgang H. Polak. *Quantum Computing: A Gentle Introduction.* MIT Press, 2011.

[59] Ronald Rivest, Adi Shamir, and Leonard Adleman. A method of obtaining digital signatures and public-key cryptosystems. *Communications of ACM*, 21:120–126, 1977.

[60] Jun John Sakurai. *Modern Quantum Mechanics.* Addison-Wesley, 1995.

[61] Erwin Schrodinger. The present situation in quantum mechanics. In *Section I.11 of Part I of Quantum Theory and Measurement.* Princeton university Press, 1983. Original German in Die Naturwissenschaften, Volume 23, Issue 48, pp. 807-812, 1935.

[62] Ramamurti Shankar. *Principles of Quantum Mechanics.* Springer, 2 edition, 1980.

[63] Claude E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27:379423, 623656, 1948.

[64] Abner Shimony. Bell's theorem. In E. N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*, volume Summer 2005. http://plato.stanford.edu/archives/sum2005/entries/bell-theorem/, 2005.

[65] Daniel R. Simon. On the power of quantum computation. *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, page 116123, 1994.

[66] Simon Singh. *The Code Book: The Science of Secrecy from Ancient Egypt to Quantum Cryptography.* Anchor, 2000.

[67] Joachim Stolze and Dieter Suter. *Quantum Computing.* Wiley-VCH, 2008.

[68] Leo Szilard. On the decrease of entropy in a thermodynamic system by the intervention of intelligent beings. In *Section V.1 of Part I of Quantum Theory and Measurement.* Princeton University Press, 1983. English translation of original published in Zeitschrift fur Physik, 1929.

[69] Alan M. Turing. On computable numbers, with an application to the entscheidungsproblem. *Proceedings of the London Mathematical Society, ser. 2*, 42:230–265, 1936-37.

[70] John von Neumann. *Mathematical Foundations of Quantum Mechanics.* Princeton University Press, 1932.

[71] Andy Wehrl. General properties of entropy. *Reviews of Modern Physics*, 50:221, 1978.

[72] John A. Wheeler and W. H. Zurek. *Quantum Theory and Measurement.* Princeton University Press, 1983.

[73] Mark Wilde. *From Classical to Quantum Shannon Theory.* Online under a Creative Commons License, 2012.

[74] Colin P Willams and Scott H Clearwater. *Explorations in Quantum Computing.* Springer, TELOS, 1997.

[75] William Wootters. Entanglement of formation of an arbitrary state of two qubits. *Physical Review Letters*, 80:2245, 1998.

[76] William Wootters and Wojciech Zurek. A single quantum cannot be cloned. *Nature*, 299:802803, 1982.

[77] Noson Yanofsky and Mirco Mannucci. *Quantum Computing for Computer Scientists.* Cambridge University Press, 2008.

Physics

Introduction to Quantum Physics and Information Processing guides you in understanding the current state of research in the novel, interdisciplinary area of quantum information. The book goes deeply into issues of quantum theory without raising the technical level too much.

The text begins with the basics of quantum mechanics required to understand how two-level systems are used as qubits. It goes on to show how quantum properties are exploited in devising algorithms for problems that are more efficient than the classical counterpart. It then explores more sophisticated notions that form the backbone of quantum information theory.

Features

- Presents important fundamental ideas of quantum information science
- Emphasizes the true meaning of the quantum mechanical description of nature
- Introduces the methods, notation, and theoretical framework of quantum mechanics
- Describes basic algorithms used in quantum computation, such as the Deutsch–Josza, Grover, and Fourier transform-based algorithms
- Addresses the notion of information content in qubits, cryptographic applications of quantum information processing, and quantum error correction
- Includes examples, exercises, problems, and references in each chapter that encourage hands-on practice and further exploration

Requiring no background in quantum physics, this text prepares you to follow more advanced books and research material in this rapidly growing field. Examples, detailed discussions, exercises, and problems facilitate a thorough, real-world understanding of quantum information.

CRC Press
Taylor & Francis Group
an **informa** business
www.crcpress.com

6000 Broken Sound Parkway, NW
Suite 300, Boca Raton, FL 33487
711 Third Avenue
New York, NY 10017
2 Park Square, Milton Park
Abingdon, Oxon OX14 4RN, UK

K23207
ISBN: 978-1-4822-3811-2
90000
9 781482 238112
www.crcpress.com