Nomenclature and notation

There are several items of nomenclature and notation which have two or more meanings in common use in the field of quantum computation and quantum information. To prevent confusion from arising, this section collects many of the more frequently used of these items, together with the conventions that will be adhered to in this book.

Linear algebra and quantum mechanics

All vector spaces are assumed to be finite dimensional, unless otherwise noted. In many instances this restriction is unnecessary, or can be removed with some additional technical work, but making the restriction globally makes the presentation more easily comprehensible, and doesn't detract much from many of the intended applications of the results.

A *positive* operator A is one for which $\langle\psi|A|\psi\rangle \geq 0$ for all $|\psi\rangle$. A *positive definite* operator A is one for which $\langle\psi|A|\psi\rangle > 0$ for all $|\psi\rangle \neq 0$. The *support* of an operator is defined to be the vector space orthogonal to its kernel. For a Hermitian operator, this means the vector space spanned by eigenvectors of the operator with non-zero eigenvalues.

The notation U (and often but not always V) will generically be used to denote a unitary operator or matrix. H is usually used to denote a quantum logic gate, the *Hadamard gate*, and sometimes to denote the *Hamiltonian* for a quantum system, with the meaning clear from context.

Vectors will sometimes be written in column format, as for example,

$$\begin{bmatrix} 1 \\ 2 \end{bmatrix}, \tag{0.1}$$

and sometimes for readability in the format $(1, 2)$. The latter should be understood as shorthand for a column vector. For two-level quantum systems used as qubits, we shall usually identify the state $|0\rangle$ with the vector $(1, 0)$, and similarly $|1\rangle$ with $(0, 1)$. We also define the Pauli sigma matrices in the conventional way – see 'Frequently used quantum gates and circuit symbols', below. Most significantly, the convention for the Pauli sigma z matrix is that $\sigma_z|0\rangle = |0\rangle$ and $\sigma_z|1\rangle = -|1\rangle$, which is reverse of what some physicists (but usually not computer scientists or mathematicians) intuitively expect. The origin of this dissonance is that the $+1$ eigenstate of σ_z is often identified by physicists with a so-called 'excited state', and it seems natural to many to identify this with $|1\rangle$, rather than with $|0\rangle$ as is done in this book. Our choice is made in order to be consistent with the usual indexing of matrix elements in linear algebra, which makes it natural to identify the first column of σ_z with the action of σ_z on $|0\rangle$, and the second column with the action on $|1\rangle$. This choice is also in use throughout the quantum computation and quantum information community. In addition to the conventional notations σ_x, σ_y and σ_z for the Pauli sigma matrices, it will also be convenient to use the notations $\sigma_1, \sigma_2, \sigma_3$ for these three matrices, and to define σ_0 as the 2×2 identity matrix. Most often, however, we use the notations I, X, Y and Z for $\sigma_0, \sigma_1, \sigma_2$ and σ_3, respectively.

Information theory and probability

As befits good information theorists, logarithms are *always* taken to base two, unless otherwise noted. We use $\log(x)$ to denote logarithms to base 2, and $\ln(x)$ on those rare occasions when we wish to take a natural logarithm. The term ***probability distribution*** is used to refer to a finite set of real numbers, p_x, such that $p_x \geq 0$ and $\sum_x p_x = 1$. The *relative entropy* of a positive operator A with respect to a positive operator B is defined by $S(A||B) \equiv \mathrm{tr}(A \log A) - \mathrm{tr}(A \log B)$.

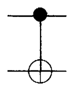
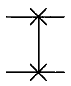
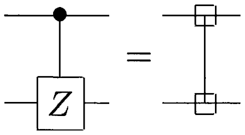
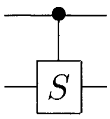
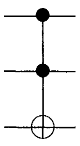
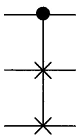
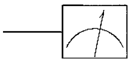
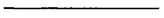
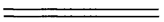
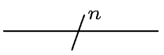
Miscellanea

$\oplus$ denotes modulo two addition. Throughout this book 'z' is pronounced 'zed'.

Frequently used quantum gates and circuit symbols

Certain schematic symbols are often used to denote unitary transforms which are useful in the design of quantum circuits. For the reader's convenience, many of these are gathered together below. The rows and columns of the unitary transforms are labeled from left to right and top to bottom as $00\ldots0, 00\ldots1$ to $11\ldots1$ with the bottom-most wire being the least significant bit. Note that $e^{i\pi/4}$ is the square root of i, so that the $\pi/8$ gate is the square root of the phase gate, which itself is the square root of the Pauli-Z gate.

Hadamard	$-\boxed{H}-$	$\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$
Pauli-X	$-\boxed{X}-$	$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
Pauli-Y	$-\boxed{Y}-$	$\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$
Pauli-Z	$-\boxed{Z}-$	$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$
Phase	$-\boxed{S}-$	$\begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}$
$\pi/8$	$-\boxed{T}-$	$\begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix}$

controlled-NOT		$\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{bmatrix}$
swap		$\begin{bmatrix}1&0&0&0\\0&0&1&0\\0&1&0&0\\0&0&0&1\end{bmatrix}$
controlled-Z		$\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&-1\end{bmatrix}$
controlled-phase		$\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&i\end{bmatrix}$
Toffoli		$\begin{bmatrix}1&0&0&0&0&0&0&0\\0&1&0&0&0&0&0&0\\0&0&1&0&0&0&0&0\\0&0&0&1&0&0&0&0\\0&0&0&0&1&0&0&0\\0&0&0&0&0&1&0&0\\0&0&0&0&0&0&0&1\\0&0&0&0&0&0&1&0\end{bmatrix}$
Fredkin (controlled-swap)		$\begin{bmatrix}1&0&0&0&0&0&0&0\\0&1&0&0&0&0&0&0\\0&0&1&0&0&0&0&0\\0&0&0&1&0&0&0&0\\0&0&0&0&1&0&0&0\\0&0&0&0&0&0&1&0\\0&0&0&0&0&1&0&0\\0&0&0&0&0&0&0&1\end{bmatrix}$
measurement		Projection onto $\|0\rangle$ and $\|1\rangle$
qubit		wire carrying a single qubit (time goes left to right)
classical bit		wire carrying a single classical bit
n qubits		wire carrying n qubits

I Fundamental concepts

1 Introduction and overview

Science offers the boldest metaphysics of the age. It is a thoroughly human construct, driven by the faith that if we dream, press to discover, explain, and dream again, thereby plunging repeatedly into new terrain, the world will somehow come clearer and we will grasp the true strangeness of the universe. And the strangeness will all prove to be connected, and make sense.
– Edward O. Wilson

Information is physical.
– Rolf Landauer

What are the fundamental concepts of quantum computation and quantum information? How did these concepts develop? To what uses may they be put? How will they be presented in this book? The purpose of this introductory chapter is to answer these questions by developing in broad brushstrokes a picture of the field of quantum computation and quantum information. The intent is to communicate a basic understanding of the central concepts of the field, perspective on how they have been developed, and to help you decide how to approach the rest of the book.

Our story begins in Section 1.1 with an account of the historical context in which quantum computation and quantum information has developed. Each remaining section in the chapter gives a brief introduction to one or more fundamental concepts from the field: quantum bits (Section 1.2), quantum computers, quantum gates and quantum circuits (Section 1.3), quantum algorithms (Section 1.4), experimental quantum information processing (Section 1.5), and quantum information and communication (Section 1.6).

Along the way, illustrative and easily accessible developments such as quantum teleportation and some simple quantum algorithms are given, using the basic mathematics taught in this chapter. The presentation is self-contained, and designed tô be accessible even without a background in computer science or physics. As we move along, we give pointers to more in-depth discussions in later chapters, where references and suggestions for further reading may also be found.

If as you read you're finding the going rough, skip on to a spot where you feel more comfortable. At points we haven't been able to avoid using a little technical lingo which won't be completely explained until later in the book. Simply accept it for now, and come back later when you understand all the terminology in more detail. The emphasis in this first chapter is on the big picture, with the details to be filled in later.

1.1 Global perspectives

Quantum computation and quantum information is the study of the information processing tasks that can be accomplished using quantum mechanical systems. Sounds pretty simple and obvious, doesn't it? Like many simple but profound ideas it was a long time before anybody thought of doing information processing using quantum mechanical systems. To see why this is the case, we must go back in time and look in turn at each of the fields which have contributed fundamental ideas to quantum computation and quantum information – quantum mechanics, computer science, information theory, and cryptography. As we take our short historical tour of these fields, think of yourself first as a physicist, then as a computer scientist, then as an information theorist, and finally as a cryptographer, in order to get some feel for the disparate perspectives which have come together in quantum computation and quantum information.

1.1.1 History of quantum computation and quantum information

Our story begins at the turn of the twentieth century when a unheralded revolution was underway in science. A series of crises had arisen in physics. The problem was that the theories of physics at that time (now dubbed *classical physics*) were predicting absurdities such as the existence of an 'ultraviolet catastrophe' involving infinite energies, or electrons spiraling inexorably into the atomic nucleus. At first such problems were resolved with the addition of *ad hoc* hypotheses to classical physics, but as a better understanding of atoms and radiation was gained these attempted explanations became more and more convoluted. The crisis came to a head in the early 1920s after a quarter century of turmoil, and resulted in the creation of the modern theory of *quantum mechanics*. Quantum mechanics has been an indispensable part of science ever since, and has been applied with enormous success to everything under and inside the Sun, including the structure of the atom, nuclear fusion in stars, superconductors, the structure of DNA, and the elementary particles of Nature.

What is quantum mechanics? Quantum mechanics is a mathematical framework or set of rules for the construction of physical theories. For example, there is a physical theory known as *quantum electrodynamics* which describes with fantastic accuracy the interaction of atoms and light. Quantum electrodynamics is built up within the framework of quantum mechanics, but it contains specific rules not determined by quantum mechanics. The relationship of quantum mechanics to specific physical theories like quantum electrodynamics is rather like the relationship of a computer's operating system to specific applications software – the operating system sets certain basic parameters and modes of operation, but leaves open how specific tasks are accomplished by the applications.

The rules of quantum mechanics are simple but even experts find them counterintuitive, and the earliest antecedents of quantum computation and quantum information may be found in the long-standing desire of physicists to better understand quantum mechanics. The best known critic of quantum mechanics, Albert Einstein, went to his grave unreconciled with the theory he helped invent. Generations of physicists since have wrestled with quantum mechanics in an effort to make its predictions more palatable. One of the goals of quantum computation and quantum information is to develop tools which sharpen our intuition about quantum mechanics, and make its predictions more transparent to human minds.

For example, in the early 1980s, interest arose in whether it might be possible to use quantum effects to signal faster than light – a big no-no according to Einstein's theory of relativity. The resolution of this problem turns out to hinge on whether it is possible to *clone* an unknown quantum state, that is, construct a copy of a quantum state. If cloning were possible, then it would be possible to signal faster than light using quantum effects.

However, cloning – so easy to accomplish with classical information (consider the words in front of you, and where they came from!) – turns out not to be possible in general in quantum mechanics. This *no-cloning theorem*, discovered in the early 1980s, is one of the earliest results of quantum computation and quantum information. Many refinements of the no-cloning theorem have since been developed, and we now have conceptual tools which allow us to understand how well a (necessarily imperfect) quantum cloning device might work. These tools, in turn, have been applied to understand other aspects of quantum mechanics.

A related historical strand contributing to the development of quantum computation and quantum information is the interest, dating to the 1970s, of obtaining *complete control over single quantum systems*. Applications of quantum mechanics prior to the 1970s typically involved a gross level of control over a bulk sample containing an enormous number of quantum mechanical systems, none of them directly accessible. For example, superconductivity has a superb quantum mechanical explanation. However, because a superconductor involves a huge (compared to the atomic scale) sample of conducting metal, we can only probe a few aspects of its quantum mechanical nature, with the individual quantum systems constituting the superconductor remaining inaccessible. Systems such as particle accelerators do allow limited access to individual quantum systems, but again provide little control over the constituent systems.

Since the 1970s many techniques for controlling single quantum systems have been developed. For example, methods have been developed for trapping a single atom in an 'atom trap', isolating it from the rest of the world and allowing us to probe many different aspects of its behavior with incredible precision. The scanning tunneling microscope has been used to move single atoms around, creating designer arrays of atoms at will. Electronic devices whose operation involves the transfer of only single electrons have been demonstrated.

Why all this effort to attain complete control over single quantum systems? Setting aside the many technological reasons and concentrating on pure science, the principal answer is that researchers have done this on a hunch. Often the most profound insights in science come when we develop a method for probing a new regime of Nature. For example, the invention of radio astronomy in the 1930s and 1940s led to a spectacular sequence of discoveries, including the galactic core of the Milky Way galaxy, pulsars, and quasars. Low temperature physics has achieved its amazing successes by finding ways to lower the temperatures of different systems. In a similar way, by obtaining complete control over single quantum systems, we are exploring untouched regimes of Nature in the hope of discovering new and unexpected phenomena. We are just now taking our first steps along these lines, and already a few interesting surprises have been' discovered in this regime. What else shall we discover as we obtain more complete control over single quantum systems, and extend it to more complex systems?

Quantum computation and quantum information fit naturally into this program. They provide a useful series of challenges at varied levels of difficulty for people devising methods to better manipulate single quantum systems, and stimulate the development of new experimental techniques and provide guidance as to the most interesting directions in which to take experiment. Conversely, the ability to control single quantum systems is essential if we are to harness the power of quantum mechanics for applications to quantum computation and quantum information.

Despite this intense interest, efforts to build quantum information processing systems have resulted in modest success to date. Small quantum computers, capable of doing dozens of operations on a few qubits represent the state of the art in quantum computation. Experimental prototypes for doing *quantum cryptography* – a way of communicating in secret across long distances – have been demonstrated, and are even at the level where they may be useful for some real-world applications. However, it remains a great challenge to physicists and engineers of the future to develop techniques for making large-scale quantum information processing a reality.

Let us turn our attention from quantum mechanics to another of the great intellectual triumphs of the twentieth century, computer science. The origins of computer science are lost in the depths of history. For example, cuneiform tablets indicate that by the time of Hammurabi (circa 1750 B.C.) the Babylonians had developed some fairly sophisticated algorithmic ideas, and it is likely that many of those ideas date to even earlier times.

The modern incarnation of computer science was announced by the great mathematician Alan Turing in a remarkable 1936 paper. Turing developed in detail an abstract notion of what we would now call a programmable computer, a model for computation now known as the *Turing machine*, in his honor. Turing showed that there is a *Universal Turing Machine* that can be used to simulate any other Turing machine. Furthermore, he claimed that the Universal Turing Machine *completely captures* what it means to perform a task by algorithmic means. That is, if an algorithm can be performed on *any* piece of hardware (say, a modern personal computer), then there is an equivalent algorithm for a Universal Turing Machine which performs exactly the same task as the algorithm running on the personal computer. This assertion, known as the *Church–Turing thesis* in honor of Turing and another pioneer of computer science, Alonzo Church, asserts the equivalence between the physical concept of what class of algorithms can be performed on *some physical device* with the rigorous mathematical concept of a Universal Turing Machine. The broad acceptance of this thesis laid the foundation for the development of a rich theory of computer science.

Not long after Turing's paper, the first computers constructed from electronic components were developed. John von Neumann developed a simple theoretical model for how to put together in a practical fashion all the components necessary for a computer to be fully as capable as a Universal Turing Machine. Hardware development truly took off, though, in 1947, when John Bardeen, Walter Brattain, and Will Shockley developed the transistor. Computer hardware has grown in power at an amazing pace ever since, so much so that the growth was codified by Gordon Moore in 1965 in what has come to be known as *Moore's law*, which states that computer power will double for constant cost roughly once every two years.

Amazingly enough, Moore's law has approximately held true in the decades since the 1960s. Nevertheless, most observers expect that this dream run will end some time during the first two decades of the twenty-first century. Conventional approaches to the fabrication of computer technology are beginning to run up against fundamental difficulties of size. Quantum effects are beginning to interfere in the functioning of electronic devices as they are made smaller and smaller.

One possible solution to the problem posed by the eventual failure of Moore's law is to move to a different computing paradigm. One such paradigm is provided by the theory of quantum computation, which is based on the idea of using quantum mechanics to perform computations, instead of classical physics. It turns out that while an ordinary computer can be used to simulate a quantum computer, it appears to be impossible to perform the simulation in an *efficient* fashion. Thus quantum computers offer an essential speed advantage over classical computers. This speed advantage is so significant that many researchers believe that *no* conceivable amount of progress in classical computation would be able to overcome the gap between the power of a classical computer and the power of a quantum computer.

What do we mean by 'efficient' versus 'inefficient' simulations of a quantum computer? Many of the key notions needed to answer this question were actually invented before the notion of a quantum computer had even arisen. In particular, the idea of *efficient* and *inefficient* algorithms was made mathematically precise by the field of *computational complexity*. Roughly speaking, an efficient algorithm is one which runs in time polynomial in the size of the problem solved. In contrast, an inefficient algorithm requires super-polynomial (typically exponential) time. What was noticed in the late 1960s and early 1970s was that it seemed as though the Turing machine model of computation was at least as powerful as any other model of computation, in the sense that a problem which could be solved efficiently in some model of computation could also be solved efficiently in the Turing machine model, by using the Turing machine to simulate the other model of computation. This observation was codified into a strengthened version of the Church–Turing thesis:

Any algorithmic process can be simulated efficiently using a Turing machine.

The key strengthening in the strong Church–Turing thesis is the word *efficiently*. If the strong Church–Turing thesis is correct, then it implies that no matter what type of machine we use to perform our algorithms, that machine can be simulated efficiently using a standard Turing machine. This is an important strengthening, as it implies that for the purposes of analyzing whether a given computational task can be accomplished efficiently, we may restrict ourselves to the analysis of the Turing machine model of computation.

One class of challenges to the strong Church–Turing thesis comes from the field of *analog computation*. In the years since Turing, many different teams of researchers have noticed that certain types of analog computers can efficiently solve problems believed to have no efficient solution on a Turing machine. At first glance these analog computers appear to violate the strong form of the Church–Turing thesis. Unfortunately for analog computation, it turns out that when realistic assumptions about the presence of noise in analog computers are made, their power disappears in all known instances; they cannot efficiently solve problems which are not efficiently solvable on a Turing machine. This lesson – that the effects of realistic noise must be taken into account in evaluating the efficiency of a computational model – was one of the great early challenges of quantum computation and quantum information, a challenge successfully met by the development of a theory of *quantum error-correcting codes* and *fault-tolerant quantum computation*. Thus, unlike analog computation, quantum computation can in principle tolerate a finite amount of noise and still retain its computational advantages.

The first major challenge to the strong Church–Turing thesis arose in the mid 1970s, when Robert Solovay and Volker Strassen showed that it is possible to test whether an integer is prime or composite using a *randomized algorithm*. That is, the Solovay–Strassen test for primality used randomness as an *essential* part of the algorithm. The algorithm did not determine whether a given integer was prime or composite with certainty. Instead, the algorithm could determine that a number was *probably* prime or else composite *with* *certainty*. By repeating the Solovay–Strassen test a few times it is possible to determine with near certainty whether a number is prime or composite. Of especial interest at the time the Solovay–Strassen test was proposed was that no efficient deterministic test for primality was known. Thus, it seemed as though computers with access to a random number generator would be able to efficiently perform computational tasks with no efficient solution on a conventional deterministic Turing machine. This discovery inspired a search for other randomized algorithms which has paid off handsomely, with the field blossoming into a thriving area of research.

Randomized algorithms pose a challenge to the strong Church–Turing thesis, suggesting that there are efficiently soluble problems which, nevertheless, cannot be efficiently solved on a deterministic Turing machine. This challenge appears to be easily resolved by a simple modification of the strong Church–Turing thesis:

Any algorithmic process can be simulated efficiently using a
probabilistic Turing machine.

This *ad hoc* modification of the strong Church–Turing thesis should leave you feeling rather queasy. Might it not turn out at some later date that yet another model of computation allows one to efficiently solve problems that are not efficiently soluble within Turing's model of computation? Is there any way we can find a single model of computation which is guaranteed to be able to efficiently simulate any other model of computation?

Motivated by this question, in 1985 David Deutsch asked whether the laws of physics could be use to *derive* an even stronger version of the Church–Turing thesis. Instead of adopting *ad hoc* hypotheses, Deutsch looked to physical theory to provide a foundation for the Church–Turing thesis that would be as secure as the status of that physical theory. In particular, Deutsch attempted to define a computational device that would be capable of efficiently simulating an *arbitrary* physical system. Because the laws of physics are ultimately quantum mechanical, Deutsch was naturally led to consider computing devices based upon the principles of quantum mechanics. These devices, quantum analogues of the machines defined forty-nine years earlier by Turing, led ultimately to the modern conception of a quantum computer used in this book.

At the time of writing it is not clear whether Deutsch's notion of a Universal Quantum Computer is sufficient to efficiently simulate an arbitrary physical system. Proving or refuting this conjecture is one of the great open problems of the field of quantum computation and quantum information. It is possible, for example, that some effect of quantum field theory or an even more esoteric effect based in string theory, quantum gravity or some other physical theory may take us beyond Deutsch's Universal Quantum Computer, giving us a still more powerful model for computation. At this stage, we simply don't know.

What Deutsch's model of a quantum computer did enable was a challenge to the strong form of the Church–Turing thesis. Deutsch asked whether it is possible for a quantum computer to efficiently solve computational problems which have no efficient solution on a classical computer, even a probabilistic Turing machine. He then constructed a simple example suggesting that, indeed, quantum computers might have computational powers exceeding those of classical computers.

This remarkable first step taken by Deutsch was improved in the subsequent decade by many people, culminating in Peter Shor's 1994 demonstration that two enormously important problems – the problem of finding the prime factors of an integer, and the so-called 'discrete logarithm' problem – could be solved efficiently on a quantum computer. This attracted widespread interest because these two problems were and still are widely believed to have no efficient solution on a classical computer. Shor's results are a powerful indication that quantum computers are more powerful than Turing machines, even probabilistic Turing machines. Further evidence for the power of quantum computers came in 1995 when Lov Grover showed that another important problem – the problem of conducting a search through some unstructured search space – could also be sped up on a quantum computer. While Grover's algorithm did not provide as spectacular a speed-up as Shor's algorithms, the widespread applicability of search-based methodologies has excited considerable interest in Grover's algorithm.

At about the same time as Shor's and Grover's algorithms were discovered, many people were developing an idea Richard Feynman had suggested in 1982. Feynman had pointed out that there seemed to be essential difficulties in simulating quantum mechanical systems on classical computers, and suggested that building computers based on the principles of quantum mechanics would allow us to avoid those difficulties. In the 1990s several teams of researchers began fleshing this idea out, showing that it is indeed possible to use quantum computers to efficiently simulate systems that have no known efficient simulation on a classical computer. It is likely that one of the major applications of quantum computers in the future will be performing simulations of quantum mechanical systems too difficult to simulate on a classical computer, a problem with profound scientific and technological implications.

What other problems can quantum computers solve more quickly than classical computers? The short answer is that we don't know. Coming up with good quantum algorithms seems to be *hard*. A pessimist might think that's because there's nothing quantum computers are good for other than the applications already discovered! We take a different view. Algorithm design for quantum computers is hard because designers face two difficult problems not faced in the construction of algorithms for classical computers. First, our human intuition is rooted in the classical world. If we use that intuition as an aid to the construction of algorithms, then the algorithmic ideas we come up with will be classical ideas. To design good quantum algorithms one must 'turn off' one's classical intuition for at least part of the design process, using truly quantum effects to achieve the desired algorithmic end. Second, to be truly interesting it is not enough to design an algorithm that is merely quantum mechanical. The algorithm must be *better* than any existing classical algorithm! Thus, it is possible that one may find an algorithm which makes use of truly quantum aspects of quantum mechanics, that is nevertheless not of widespread interest because classical algorithms with comparable performance characteristics exist. The combination of these two problems makes the construction of new quantum algorithms a challenging problem for the future.

Even more broadly, we can ask if there are any generalizations we can make about the power of quantum computers versus classical computers. What is it that makes quantum computers more powerful than classical computers – assuming that this is indeed the case? What class of problems can be solved efficiently on a quantum computer, and how does that class compare to the class of problems that can be solved efficiently on a classical computer? One of the most exciting things about quantum computation and quantum information is how *little* is known about the answers to these questions! It is a great challenge for the future to understand these questions better.

Having come up to the frontier of quantum computation, let's switch to the history of another strand of thought contributing to quantum computation and quantum information: information theory. At the same time computer science was exploding in the 1940s, another revolution was taking place in our understanding of *communication*. In 1948 Claude Shannon published a remarkable pair of papers laying the foundations for the modern theory of information and communication.

Perhaps the key step taken by Shannon was *to mathematically define the concept of information*. In many mathematical sciences there is considerable flexibility in the choice of fundamental definitions. Try thinking naively for a few minutes about the following question: how would you go about mathematically defining the notion of an information source? Several *different* answers to this problem have found widespread use; however, the definition Shannon came up with seems to be far and away the most fruitful in terms of increased understanding, leading to a plethora of deep results and a theory with a rich structure which seems to accurately reflect many (though not all) real-world communications problems.

Shannon was interested in two key questions related to the communication of information over a communications channel. First, what resources are required to send information over a communications channel? For example, telephone companies need to know how much information they can reliably transmit over a given telephone cable. Second, can information be transmitted in such a way that it is protected against noise in the communications channel?

Shannon answered these two questions by proving the two fundamental theorems of information theory. The first, Shannon's *noiseless channel coding theorem*, quantifies the physical resources required to store the output from an information source. Shannon's second fundamental theorem, the *noisy channel coding theorem*, quantifies how much information it is possible to reliably transmit through a noisy communications channel. To achieve reliable transmission in the presence of noise, Shannon showed that *error-correcting codes* could be used to protect the information being sent. Shannon's noisy channel coding theorem gives an upper limit on the protection afforded by error-correcting codes. Unfortunately, Shannon's theorem does not explicitly give a practically useful set of error-correcting codes to achieve that limit. From the time of Shannon's papers until today, researchers have constructed more and better classes of error-correcting codes in their attempts to come closer to the limit set by Shannon's theorem. A sophisticated theory of error-correcting codes now exists offering the user a plethora of choices in their quest to design a good error-correcting code. Such codes are used in a multitude of places including, for example, compact disc players, computer modems, and satellite communications systems.

Quantum information theory has followed with similar developments. In 1995, Ben Schumacher provided an analogue to Shannon's noiseless coding theorem, and in the process defined the 'quantum bit' or 'qubit' as a tangible physical resource. However, no analogue to Shannon's noisy channel coding theorem is yet known for quantum information. Nevertheless, in analogy to their classical counterparts, a theory of quantum error-correction has been developed which, as already mentioned, allows quantum computers to compute effectively in the presence of noise, and also allows communication over noisy *quantum* channels to take place reliably.

Indeed, classical ideas of error-correction have proved to be enormously important in developing and understanding quantum error-correcting codes. In 1996, two groups working independently, Robert Calderbank and Peter Shor, and Andrew Steane, discovered an important class of quantum codes now known as CSS codes after their initials. This work has since been subsumed by the stabilizer codes, independently discovered by Robert Calderbank, Eric Rains, Peter Shor and Neil Sloane, and by Daniel Gottesman. By building upon the basic ideas of classical linear coding theory, these discoveries greatly facilitated a rapid understanding of quantum error-correcting codes and their application to quantum computation and quantum information.

The theory of quantum error-correcting codes was developed to protect quantum states against noise. What about transmitting ordinary *classical* information using a quantum channel? How efficiently can this be done? A few surprises have been discovered in this arena. In 1992 Charles Bennett and Stephen Wiesner explained how to transmit *two* classical bits of information, while only transmitting *one* quantum bit from sender to receiver, a result dubbed *superdense coding*.

Even more interesting are the results in *distributed quantum computation*. Imagine you have two computers networked, trying to solve a particular problem. How much communication is required to solve the problem? Recently it has been shown that quantum computers can require *exponentially less* communication to solve certain problems than would be required if the networked computers were classical! Unfortunately, as yet these problems are not especially important in a practical setting, and suffer from some undesirable technical restrictions. A major challenge for the future of quantum computation and quantum information is to find problems of real-world importance for which distributed quantum computation offers a substantial advantage over distributed classical computation.

Let's return to information theory proper. The study of information theory begins with the properties of a single communications channel. In applications we often do not deal with a single communications channel, but rather with networks of many channels. The subject of *networked information theory* deals with the information carrying properties of such networks of communications channels, and has been developed into a rich and intricate subject.

By contrast, the study of networked quantum information theory is very much in its infancy. Even for very basic questions we know little about the information carrying abilities of networks of quantum channels. Several rather striking preliminary results have been found in the past few years; however, no unifying theory of networked information theory exists for quantum channels. One example of networked quantum information theory should suffice to convince you of the value such a general theory would have. Imagine that we are attempting to send quantum information from Alice to Bob through a noisy quantum channel. If that channel has zero capacity for quantum information, then it is impossible to reliably send *any* information from Alice to Bob. Imagine instead that we consider two copies of the channel, operating in synchrony. Intuitively it is clear (and can be rigorously justified) that such a channel also has zero capacity to send quantum information. However, if we instead *reverse* the direction of one of the channels, as illustrated in Figure 1.1, it turns out that sometimes we can obtain a non-zero capacity for the transmission of information from Alice to Bob! Counter-intuitive properties like this illustrate the strange nature of quantum information. Better understanding the information carrying properties of networks of quantum channels is a major open problem of quantum computation and quantum information.

Let's switch fields one last time, moving to the venerable old art and science of *cryptography*. Broadly speaking, cryptography is the problem of doing *communication* or

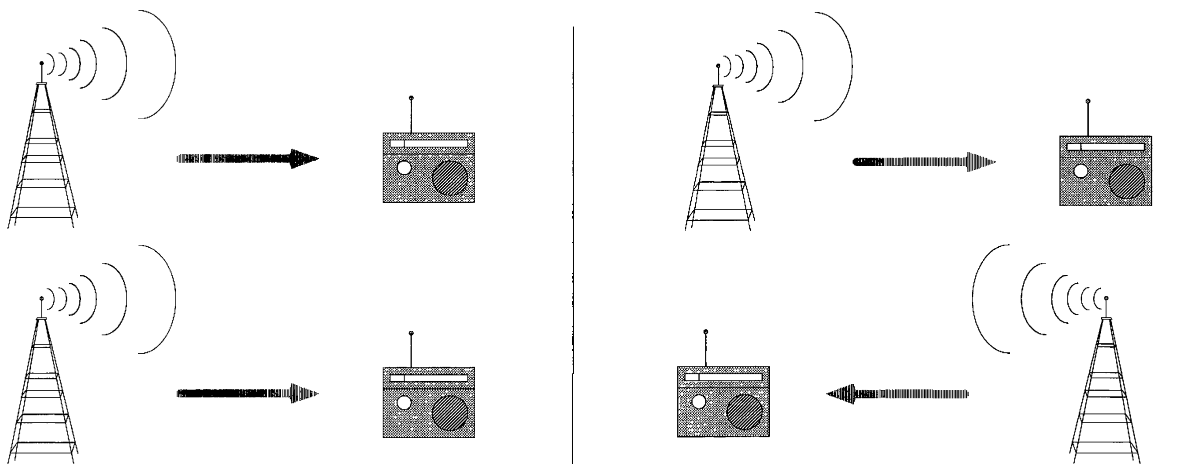


Figure 1.1. Classically, if we have two very noisy channels of zero capacity running side by side, then the combined channel has zero capacity to send information. Not surprisingly, if we reverse the direction of one of the channels, we still have zero capacity to send information. Quantum mechanically, reversing one of the zero capacity channels can actually allow us to send information!

computation involving two or more parties *who may not trust one another*. The best known cryptographic problem is the transmission of secret messages. Suppose two parties wish to communicate in secret. For example, you may wish to give your credit card number to a merchant in exchange for goods, hopefully without any malevolent third party intercepting your credit card number. The way this is done is to use a *cryptographic protocol*. We'll describe in detail how cryptographic protocols work later in the book, but for now it will suffice to make a few simple distinctions. The most important distinction is between *private key cryptosystems* and *public key cryptosystems*.

The way a private key cryptosystem works is that two parties, 'Alice' and 'Bob', wish to communicate by sharing a *private key*, which only they know. The exact form of the key doesn't matter at this point – think of a string of zeroes and ones. The point is that this key is used by Alice to *encrypt* the information she wishes to send to Bob. After Alice encrypts she sends the encrypted information to Bob, who must now recover the original information. Exactly how Alice encrypts the message *depends upon the private key*, so that to recover the original message Bob needs to know the private key, in order to undo the transformation Alice applied.

Unfortunately, private key cryptosystems have some severe problems in many contexts. The most basic problem is how to distribute the keys? In many ways, the key distribution problem is just as difficult as the original problem of communicating in private – a malevolent third party may be eavesdropping on the key distribution, and then use the intercepted key to decrypt some of the message transmission.

One of the earliest discoveries in quantum computation and quantum information was that quantum mechanics can be used to do key distribution in such a way that Alice and Bob's security can not be compromised. This procedure is known as *quantum cryptography* or *quantum key distribution*. The basic idea is to exploit the quantum mechanical principle that observation in general disturbs the system being observed. Thus, if there is an eavesdropper listening in as Alice and Bob attempt to transmit their key, the presence of the eavesdropper will be visible as a disturbance of the communications channel Alice and Bob are using to establish the key. Alice and Bob can then throw out the key bits established while the eavesdropper was listening in, and start over. The first quantum cryptographic ideas were proposed by Stephen Wiesner in the late 1960s, but unfortunately were not accepted for publication! In 1984 Charles Bennett and Gilles Brassard, building on Wiesner's earlier work, proposed a protocol using quantum mechanics to distribute keys between Alice and Bob, without any possibility of a compromise. Since then numerous quantum cryptographic protocols have been proposed, and experimental prototypes developed. At the time of this writing, the experimental prototypes are nearing the stage where they may be useful in limited-scale real-world applications.

The second major type of cryptosystem is the *public key cryptosystem.* Public key cryptosystems don't rely on Alice and Bob sharing a secret key in advance. Instead, Bob simply publishes a 'public key', *which is made available to the general public.* Alice can make use of this public key to encrypt a message which she sends to Bob. What is interesting is that a third party *cannot* use Bob's public key to decrypt the message! Strictly speaking, we shouldn't say *cannot.* Rather, the encryption transformation is chosen in a very clever and non-trivial way so that it is *extremely difficult* (though not impossible) to invert, given only knowledge of the public key. To make inversion easy, Bob has a *secret key* matched to his public key, which together enable him to *easily* perform the decryption. This secret key is not known to anybody other than Bob, who can therefore be confident that only he can read the contents of Alice's transmission, to the extent that it is unlikely that anybody else has the computational power to invert the encryption, given only the public key. Public key cryptosystems solve the key distribution problem by making it unnecessary for Alice and Bob to share a private key before communicating.

Rather remarkably, public key cryptography did not achieve widespread use until the mid-1970s, when it was proposed independently by Whitfield Diffie and Martin Hellman, and by Ralph Merkle, revolutionizing the field of cryptography. A little later, Ronald Rivest, Adi Shamir, and Leonard Adleman developed the *RSA cryptosystem*, which at the time of writing is the most widely deployed public key cryptosystem, believed to offer a fine balance of security and practical usability. In 1997 it was disclosed that these ideas – public key cryptography, the Diffie–Hellman and RSA cryptosystems – were actually invented in the late 1960s and early 1970s by researchers working at the British intelligence agency GCHQ.

The key to the security of public key cryptosystems is that it should be difficult to invert the encryption stage if only the public key is available. For example, it turns out that inverting the encryption stage of RSA is a problem closely related to factoring. Much of the presumed security of RSA comes from the belief that factoring is a problem hard to solve on a classical computer. However, Shor's fast algorithm for factoring on a quantum computer could be used to break RSA! Similarly, there are other public key cryptosystems which can be broken if a fast algorithm for solving the discrete logarithm problem – like Shor's quantum algorithm for discrete logarithm – were known. This practical application of quantum computers to the breaking of cryptographic codes has excited much of the interest in quantum computation and quantum information.

We have been looking at the historical antecedents for quantum computation and quantum information. Of course, as the field has grown and matured, it has sprouted its own subfields of research, whose antecedents lie mainly within quantum computation and quantum information.

Perhaps the most striking of these is the study of *quantum entanglement.* Entanglement is a uniquely quantum mechanical *resource* that plays a key role in many of the most interesting applications of quantum computation and quantum information; entanglement is iron to the classical world's bronze age. In recent years there has been a tremendous effort trying to better understand the properties of entanglement considered as a fundamental resource of Nature, of comparable importance to energy, information, entropy, or any other fundamental resource. Although there is as yet no complete theory of entanglement, some progress has been made in understanding this strange property of quantum mechanics. It is hoped by many researchers that further study of the properties of entanglement will yield insights that facilitate the development of new applications in quantum computation and quantum information.

1.1.2 Future directions

We've looked at some at the history and present status of quantum computation and quantum information. What of the future? What can quantum computation and quantum information offer to science, to technology, and to humanity? What benefits does quantum computation and quantum information confer upon its parent fields of computer science, information theory, and physics? What are the key open problems of quantum computation and quantum information? We will make a few very brief remarks about these overarching questions before moving onto more detailed investigations.

Quantum computation and quantum information has taught us to *think physically about computation*, and we have discovered that this approach yields many new and exciting capabilities for information processing and communication. Computer scientists and information theorists have been gifted with a new and rich paradigm for exploration. Indeed, in the broadest terms we have learned that *any physical theory*, not just quantum mechanics, may be used as the basis for a theory of information processing and communication. The fruits of these explorations may one day result in information processing devices with capabilities far beyond today's computing and communications systems, with concomitant benefits and drawbacks for society as a whole.

Quantum computation and quantum information certainly offer challenges aplenty to physicists, but it is perhaps a little subtle what quantum computation and quantum information offers to physics in the long term. We believe that just as we have learned to think physically about computation, we can also learn to *think computationally about physics*. Whereas physics has traditionally been a discipline focused on understanding 'elementary' objects and simple systems, many interesting aspects of Nature arise only when things become larger and more complicated. Chemistry and engineering deal with such complexity to some extent, but most often in a rather *ad hoc* fashion. One of the messages of quantum computation and information is that new tools are available for traversing the gulf between the small and the relatively complex: computation and algorithms provide systematic means for constructing and understanding such systems. Applying ideas from these fields is already beginning to yield new insights into physics. It is our hope that this perspective will blossom in years to come into a fruitful way of understanding all aspects of physics.

We've briefly examined some of the key motivations and ideas underlying quantum computation and quantum information. Over the remaining sections of this chapter we give a more technical but still accessible introduction to these motivations and ideas, with the hope of giving you a bird's-eye view of the field as it is presently poised.

1.2 Quantum bits

The *bit* is the fundamental concept of classical computation and classical information. Quantum computation and quantum information are built upon an analogous concept, the *quantum bit*, or *qubit* for short. In this section we introduce the properties of single and multiple qubits, comparing and contrasting their properties to those of classical bits.

What is a qubit? We're going to describe qubits as *mathematical objects* with certain specific properties. 'But hang on', you say, 'I thought qubits were physical objects.' It's true that qubits, like bits, are realized as actual physical systems, and in Section 1.5 and Chapter 7 we describe in detail how this connection between the abstract mathematical point of view and real systems is made. However, for the most part we treat qubits as abstract mathematical objects. The beauty of treating qubits as abstract entities is that it gives us the freedom to construct a general theory of quantum computation and quantum information which does not depend upon a specific system for its realization.

What then is a qubit? Just as a classical bit has a *state* – either 0 or 1 – a qubit also has a state. Two possible states for a qubit are the states $|0\rangle$ and $|1\rangle$, which as you might guess correspond to the states 0 and 1 for a classical bit. Notation like '$|\ \rangle$' is called the *Dirac notation*, and we'll be seeing it often, as it's the standard notation for states in quantum mechanics. The difference between bits and qubits is that a qubit can be in a state *other* than $|0\rangle$ or $|1\rangle$. It is also possible to form *linear combinations* of states, often called *superpositions*:

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

The numbers α and β are complex numbers, although for many purposes not much is lost by thinking of them as real numbers. Put another way, the state of a qubit is a vector in a two-dimensional complex vector space. The special states $|0\rangle$ and $|1\rangle$ are known as *computational basis states*, and form an orthonormal basis for this vector space.

We can examine a bit to determine whether it is in the state 0 or 1. For example, computers do this all the time when they retrieve the contents of their memory. Rather remarkably, we cannot examine a qubit to determine its quantum state, that is, the values of α and β. Instead, quantum mechanics tells us that we can only acquire much more restricted information about the quantum state. When we measure a qubit we get either the result 0, with probability $|\alpha|^2$, or the result 1, with probability $|\beta|^2$. Naturally, $|\alpha|^2 + |\beta|^2 = 1$, since the probabilities must sum to one. Geometrically, we can interpret this as the condition that the qubit's state be normalized to length 1. Thus, in general a qubit's state is a unit vector in a two-dimensional complex vector space.

This dichotomy between the unobservable state of a qubit and the observations we can make lies at the heart of quantum computation and quantum information. In most of our abstract models of the world, there is a direct correspondence between elements of the abstraction and the real world, just as an architect's plans for a building are in correspondence with the final building. The lack of this direct correspondence in quantum mechanics makes it difficult to intuit the behavior of quantum systems; however, there is an indirect correspondence, for qubit states can be manipulated and transformed in ways which lead to measurement outcomes which depend distinctly on the different properties of the state. Thus, these quantum states have real, experimentally verifiable consequences, which we shall see are essential to the power of quantum computation and quantum information.

The ability of a qubit to be in a superposition state runs counter to our 'common sense' understanding of the physical world around us. A classical bit is like a coin: either heads or tails up. For imperfect coins, there may be intermediate states like having it balanced on an edge, but those can be disregarded in the ideal case. By contrast, a qubit can exist in a *continuum* of states between $|0\rangle$ and $|1\rangle$ – until it is observed. Let us emphasize again that when a qubit is measured, it only ever gives '0' or '1' as the measurement result – probabilistically. For example, a qubit can be in the state

$$\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle , \tag{1.2}$$

which, when measured, gives the result 0 fifty percent ($|1/\sqrt{2}|^2$) of the time, and the result 1 fifty percent of the time. We will return often to this state, which is sometimes denoted $|+\rangle$.

Despite this strangeness, qubits are decidedly real, their existence and behavior extensively validated by experiments (discussed in Section 1.5 and Chapter 7), and many different physical systems can be used to realize qubits. To get a concrete feel for how a qubit can be realized it may be helpful to list some of the ways this realization may occur: as the two different polarizations of a photon; as the alignment of a nuclear spin in a uniform magnetic field; as two states of an electron orbiting a single atom such as shown in Figure 1.2. In the atom model, the electron can exist in either the so-called 'ground' or 'excited' states, which we'll call $|0\rangle$ and $|1\rangle$, respectively. By shining light on the atom, with appropriate energy and for an appropriate length of time, it is possible to move the electron from the $|0\rangle$ state to the $|1\rangle$ state and vice versa. But more interestingly, by reducing the time we shine the light, an electron initially in the state $|0\rangle$ can be moved 'halfway' between $|0\rangle$ and $|1\rangle$, into the $|+\rangle$ state.

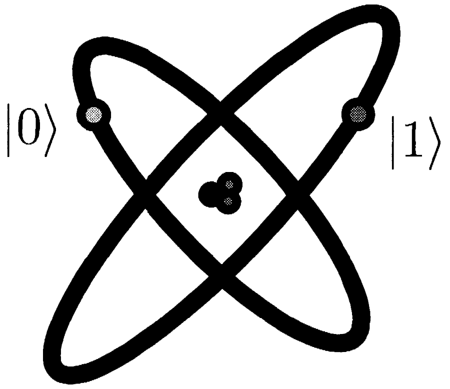


Figure 1.2. Qubit represented by two electronic levels in an atom.

Naturally, a great deal of attention has been given to the 'meaning' or 'interpretation' that might be attached to superposition states, and of the inherently probabilistic nature of observations on quantum systems. However, by and large, we shall not concern ourselves with such discussions in this book. Instead, our intent will be to develop mathematical and conceptual pictures which are predictive.

One picture useful in thinking about qubits is the following geometric representation.

Because $|\alpha|^2 + |\beta|^2 = 1$, we may rewrite Equation (1.1) as

$$|\psi\rangle = e^{i\gamma}\left(\cos\frac{\theta}{2}|0\rangle + e^{i\varphi}\sin\frac{\theta}{2}|1\rangle\right), \tag{1.3}$$

where θ, φ and γ are real numbers. In Chapter 2 we will see that we can *ignore* the factor of $e^{i\gamma}$ out the front, because it has *no observable effects*, and for that reason we can effectively write

$$|\psi\rangle = \cos\frac{\theta}{2}|0\rangle + e^{i\varphi}\sin\frac{\theta}{2}|1\rangle. \tag{1.4}$$

The numbers θ and φ define a point on the unit three-dimensional sphere, as shown in Figure 1.3. This sphere is often called the *Bloch sphere*; it provides a useful means of visualizing the state of a single qubit, and often serves as an excellent testbed for ideas about quantum computation and quantum information. Many of the operations on single qubits which we describe later in this chapter are neatly described within the Bloch sphere picture. However, it must be kept in mind that this intuition is limited because there is no simple generalization of the Bloch sphere known for multiple qubits.

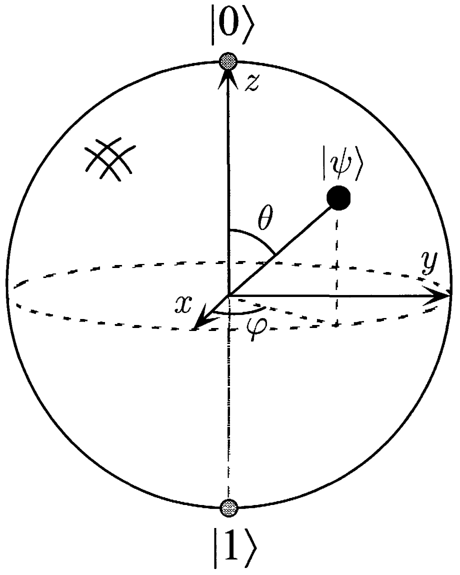


Figure 1.3. Bloch sphere representation of a qubit.

How much information is represented by a qubit? Paradoxically, there are an infinite number of points on the unit sphere, so that in principle one could store an entire text of Shakespeare in the infinite binary expansion of θ. However, this conclusion turns out to be misleading, because of the behavior of a qubit when observed. Recall that measurement of a qubit will give *only* either 0 or 1. Furthermore, measurement *changes* the state of a qubit, collapsing it from its superposition of $|0\rangle$ and $|1\rangle$ to the specific state consistent with the measurement result. For example, if measurement of $|+\rangle$ gives 0, then the post-measurement state of the qubit will be $|0\rangle$. Why does this type of collapse occur? Nobody knows. As discussed in Chapter 2, this behavior is simply one of the *fundamental postulates* of quantum mechanics. What is relevant for our purposes is that from a single measurement one obtains only a single bit of information about the state of the qubit, thus resolving the apparent paradox. It turns out that only if infinitely many identically prepared qubits were measured would one be able to determine α and β for a qubit in the state given in Equation (1.1).

But an even more interesting question to ask might be: how much information is represented by a qubit *if we do not measure it*? This is a trick question, because how can one quantify information if it cannot be measured? Nevertheless, there is something conceptually important here, because when Nature evolves a closed quantum system of qubits, not performing any 'measurements', she apparently does keep track of all the continuous variables describing the state, like α and β. In a sense, in the state of a qubit, Nature conceals a great deal of 'hidden information'. And even more interestingly, we will see shortly that the potential amount of this extra 'information' grows exponentially with the number of qubits. Understanding this hidden *quantum information* is a question that we grapple with for much of this book, and which lies at the heart of what makes quantum mechanics a powerful tool for information processing.

1.2.1 Multiple qubits

> *Hilbert space is a big place.*
> – Carlton Caves

Suppose we have two qubits. If these were two classical bits, then there would be four possible states, 00, 01, 10, and 11. Correspondingly, a two qubit system has four *computational basis states* denoted $|00\rangle, |01\rangle, |10\rangle, |11\rangle$. A pair of qubits can also exist in superpositions of these four states, so the quantum state of two qubits involves associating a complex coefficient – sometimes called an *amplitude* – with each computational basis state, such that the state vector describing the two qubits is

$$|\psi\rangle = \alpha_{00}|00\rangle + \alpha_{01}|01\rangle + \alpha_{10}|10\rangle + \alpha_{11}|11\rangle. \tag{1.5}$$

Similar to the case for a single qubit, the measurement result x ($= 00, 01, 10$ or 11) occurs with probability $|\alpha_x|^2$, with the state of the qubits after the measurement being $|x\rangle$. The condition that probabilities sum to one is therefore expressed by the *normalization* condition that $\sum_{x\in\{0,1\}^2} |\alpha_x|^2 = 1$, where the notation '$\{0,1\}^2$' means 'the set of strings of length two with each letter being either zero or one'. For a two qubit system, we could measure just a subset of the qubits, say the first qubit, and you can probably guess how this works: measuring the first qubit alone gives 0 with probability $|\alpha_{00}|^2 + |\alpha_{01}|^2$, leaving the post-measurement state

$$|\psi'\rangle = \frac{\alpha_{00}|00\rangle + \alpha_{01}|01\rangle}{\sqrt{|\alpha_{00}|^2 + |\alpha_{01}|^2}}. \tag{1.6}$$

Note how the post-measurement state is *re-normalized* by the factor $\sqrt{|\alpha_{00}|^2 + |\alpha_{01}|^2}$ so that it still satisfies the normalization condition, just as we expect for a legitimate quantum state.

An important two qubit state is the *Bell state* or *EPR pair*,

$$\frac{|00\rangle + |11\rangle}{\sqrt{2}}. \tag{1.7}$$

This innocuous-looking state is responsible for many surprises in quantum computation and quantum information. It is the key ingredient in quantum teleportation and superdense coding, which we'll come to in Section 1.3.7 and Section 2.3, respectively, and the prototype for many other interesting quantum states. The Bell state has the property that upon measuring the first qubit, one obtains two possible results: 0 with probability 1/2, leaving the post-measurement state $|\varphi'\rangle = |00\rangle$, and 1 with probability 1/2, leaving $|\varphi'\rangle = |11\rangle$. As a result, a measurement of the second qubit always gives the same result as the measurement of the first qubit. That is, the measurement outcomes are *correlated.* Indeed, it turns out that other types of measurements can be performed on the Bell state, by first applying some operations to the first or second qubit, and that interesting correlations still exist between the result of a measurement on the first and second qubit. These correlations have been the subject of intense interest ever since a famous paper by Einstein, Podolsky and Rosen, in which they first pointed out the strange properties of states like the Bell state. EPR's insights were taken up and greatly improved by John Bell, who proved an amazing result: the measurement correlations in the Bell state are *stronger than could ever exist between classical systems.* These results, described in detail in Section 2.6, were the first intimation that quantum mechanics allows information processing beyond what is possible in the classical world.

More generally, we may consider a system of n qubits. The computational basis states of this system are of the form $|x_1x_2 \ldots x_n\rangle$, and so a quantum state of such a system is specified by 2^n amplitudes. For $n = 500$ this number is larger than the estimated number of atoms in the Universe! Trying to store all these complex numbers would not be possible on any conceivable classical computer. Hilbert space is indeed a big place. In principle, however, Nature manipulates such enormous quantities of data, even for systems containing only a few hundred atoms. It is as if Nature were keeping 2^{500} hidden pieces of scratch paper on the side, on which she performs her calculations as the system evolves. This enormous potential computational power is something we would very much like to take advantage of. But how can we think of quantum mechanics as computation?

1.3 Quantum computation

Changes occurring to a quantum state can be described using the language of *quantum computation.* Analogous to the way a classical computer is built from an electrical circuit containing wires and logic gates, a quantum computer is built from a *quantum circuit* containing wires and elementary *quantum gates* to carry around and manipulate the quantum information. In this section we describe some simple quantum gates, and present several example circuits illustrating their application, including a circuit which teleports qubits!

1.3.1 Single qubit gates

Classical computer circuits consist of *wires* and *logic gates.* The wires are used to carry information around the circuit, while the logic gates perform manipulations of the information, converting it from one form to another. Consider, for example, classical single bit logic gates. The only non-trivial member of this class is the NOT gate, whose operation is defined by its *truth table*, in which 0 → 1 and 1 → 0, that is, the 0 and 1 states are interchanged.

Can an analogous quantum NOT gate for qubits be defined? Imagine that we had some process which took the state $|0\rangle$ to the state $|1\rangle$, and vice versa. Such a process would obviously be a good candidate for a quantum analogue to the NOT gate. However, specifying the action of the gate on the states $|0\rangle$ and $|1\rangle$ does not tell us what happens to superpositions of the states $|0\rangle$ and $|1\rangle$, without further knowledge about the properties of quantum gates. In fact, the quantum NOT gate acts *linearly*, that is, it takes the state

$$\alpha|0\rangle + \beta|1\rangle \tag{1.8}$$

to the corresponding state in which the role of $|0\rangle$ and $|1\rangle$ have been interchanged,

$$\alpha|1\rangle + \beta|0\rangle. \tag{1.9}$$

Why the quantum NOT gate acts linearly and not in some nonlinear fashion is a very interesting question, and the answer is not at all obvious. It turns out that this linear behavior is a general property of quantum mechanics, and very well motivated empirically; moreover, nonlinear behavior can lead to apparent paradoxes such as time travel, faster-than-light communication, and violations of the second laws of thermodynamics. We'll explore this point in more depth in later chapters, but for now we'll just take it as given.

There is a convenient way of representing the quantum NOT gate in matrix form, which follows directly from the linearity of quantum gates. Suppose we define a matrix X to represent the quantum NOT gate as follows:

$$X \equiv \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}. \tag{1.10}$$

(The notation X for the quantum NOT is used for historical reasons.) If the quantum state $\alpha|0\rangle + \beta|1\rangle$ is written in a vector notation as

$$\begin{bmatrix} \alpha \\ \beta \end{bmatrix}, \tag{1.11}$$

with the top entry corresponding to the amplitude for $|0\rangle$ and the bottom entry the amplitude for $|1\rangle$, then the corresponding output from the quantum NOT gate is

$$X \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \beta \\ \alpha \end{bmatrix}. \tag{1.12}$$

Notice that the action of the NOT gate is to take the state $|0\rangle$ and replace it by the state corresponding to the first column of the matrix X. Similarly, the state $|1\rangle$ is replaced by the state corresponding to the second column of the matrix X.

So quantum gates on a single qubit can be described by two by two matrices. Are there any constraints on what matrices may be used as quantum gates? It turns out that there are. Recall that the normalization condition requires $|\alpha|^2 + |\beta|^2 = 1$ for a quantum state $\alpha|0\rangle + \beta|1\rangle$. This must also be true of the quantum state $|\psi'\rangle = \alpha'|0\rangle + \beta'|1\rangle$ after the gate has acted. It turns out that the appropriate condition on the matrix representing the gate is that the matrix U describing the single qubit gate be *unitary*, that is $U^\dagger U = I$, where $U^\dagger$ is the *adjoint* of U (obtained by transposing and then complex conjugating U), and I is the two by two identity matrix. For example, for the NOT gate it is easy to verify that $X^\dagger X = I$.

Amazingly, this *unitarity* constraint is the *only* constraint on quantum gates. Any unitary matrix specifies a valid quantum gate! The interesting implication is that in contrast to the classical case, where only one non-trivial single bit gate exists – the NOT

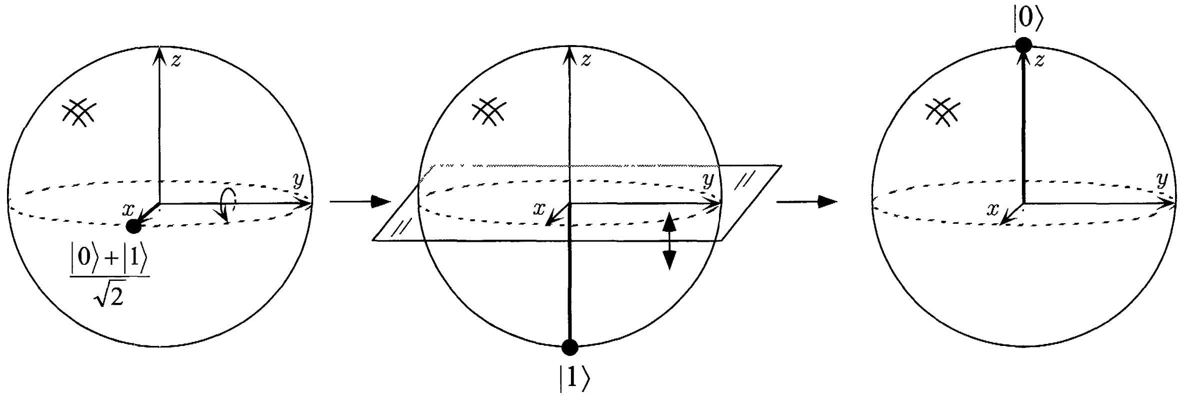


Figure 1.4. Visualization of the Hadamard gate on the Bloch sphere, acting on the input state $(|0\rangle + |1\rangle)/\sqrt{2}$.

gate – there are many non-trivial single qubit gates. Two important ones which we shall use later are the Z gate:

$$Z \equiv \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \tag{1.13}$$

which leaves $|0\rangle$ unchanged, and flips the sign of $|1\rangle$ to give $-|1\rangle$, and the *Hadamard* gate,

$$H \equiv \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}. \tag{1.14}$$

This gate is sometimes described as being like a 'square-root of NOT' gate, in that it turns a $|0\rangle$ into $(|0\rangle + |1\rangle)/\sqrt{2}$ (first column of H), 'halfway' between $|0\rangle$ and $|1\rangle$, and turns $|1\rangle$ into $(|0\rangle - |1\rangle)/\sqrt{2}$ (second column of H), which is also 'halfway' between $|0\rangle$ and $|1\rangle$. Note, however, that H^2 is not a NOT gate, as simple algebra shows that $H^2 = I$, and thus applying H twice to a state does nothing to it.

The Hadamard gate is one of the most useful quantum gates, and it is worth trying to visualize its operation by considering the Bloch sphere picture. In this picture, it turns out that single qubit gates correspond to rotations and reflections of the sphere. The Hadamard operation is just a rotation of the sphere about the $\hat{y}$ axis by 90°, followed by a reflection through the $\hat{x}$-$\hat{y}$ plane, as illustrated in Figure 1.4. Some important single qubit gates are shown in Figure 1.5, and contrasted with the classical case.

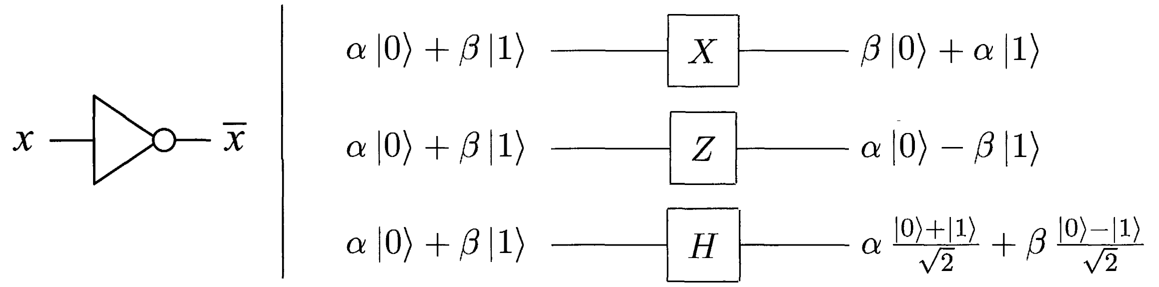

x
$\overline{x}$
$\alpha|0\rangle + \beta|1\rangle$
X
$\beta|0\rangle + \alpha|1\rangle$
$\alpha|0\rangle + \beta|1\rangle$
Z
$\alpha|0\rangle - \beta|1\rangle$
$\alpha|0\rangle + \beta|1\rangle$
H
$\alpha\frac{|0\rangle+|1\rangle}{\sqrt{2}} + \beta\frac{|0\rangle-|1\rangle}{\sqrt{2}}$

Figure 1.5. Single bit (left) and qubit (right) logic gates.

There are infinitely many two by two unitary matrices, and thus infinitely many single qubit gates. However, it turns out that the properties of the complete set can be understood from the properties of a much smaller set. For example, as explained in Box 1.1, an arbitrary single qubit unitary gate can be decomposed as a product of rotations

$$\begin{bmatrix} \cos\frac{\gamma}{2} & -\sin\frac{\gamma}{2} \\ \sin\frac{\gamma}{2} & \cos\frac{\gamma}{2} \end{bmatrix}, \tag{1.15}$$

and a gate which we'll later understand as being a rotation about the $\hat{z}$ axis,

$$\begin{bmatrix} e^{-i\beta/2} & 0 \\ 0 & e^{i\beta/2} \end{bmatrix}, \tag{1.16}$$

together with a *(global) phase shift* – a constant multiplier of the form $e^{i\alpha}$. These gates can be broken down further – we don't need to be able to do these gates for arbitrary α, β and γ, but can build arbitrarily good approximations to such gates using only certain special *fixed* values of α, β and γ. In this way it is possible to build up an arbitrary single qubit gate using a *finite* set of quantum gates. More generally, an arbitrary quantum computation on any number of qubits can be generated by a finite set of gates that is said to be *universal* for quantum computation. To obtain such a universal set we first need to introduce some quantum gates involving multiple qubits.

Box 1.1: Decomposing single qubit operations

In Section 4.2 starting on page 174 we prove that an arbitrary 2×2 unitary matrix may be decomposed as

$$U = e^{i\alpha}\begin{bmatrix} e^{-i\beta/2} & 0 \\ 0 & e^{i\beta/2} \end{bmatrix}\begin{bmatrix} \cos\frac{\gamma}{2} & -\sin\frac{\gamma}{2} \\ \sin\frac{\gamma}{2} & \cos\frac{\gamma}{2} \end{bmatrix}, \begin{bmatrix} e^{-i\delta/2} & 0 \\ 0 & e^{i\delta/2} \end{bmatrix}, \tag{1.17}$$

where α, β, γ, and δ are real-valued. Notice that the second matrix is just an ordinary rotation. It turns out that the first and last matrices can also be understood as rotations in a different plane. This decomposition can be used to give an exact prescription for performing an *arbitrary* single qubit quantum logic gate.

1.3.2 Multiple qubit gates

Now let us generalize from one to multiple qubits. Figure 1.6 shows five notable multiple bit classical gates, the AND, OR, XOR (exclusive-OR), NAND and NOR gates. An important theoretical result is that any function on bits can be computed from the composition of NAND gates alone, which is thus known as a *universal* gate. By contrast, the XOR alone or even together with NOT is not universal. One way of seeing this is to note that applying an XOR gate does not change the total parity of the bits. As a result, any circuit involving only NOT and XOR gates will, if two inputs x and y have the same parity, give outputs with the same parity, restricting the class of functions which may be computed, and thus precluding universality.

The prototypical multi-qubit quantum logic gate is the *controlled*-NOT or CNOT gate. This gate has two input qubits, known as the *control* qubit and the *target* qubit, respectively. The circuit representation for the CNOT is shown in the top right of Figure 1.6; the top line represents the control qubit, while the bottom line represents the target

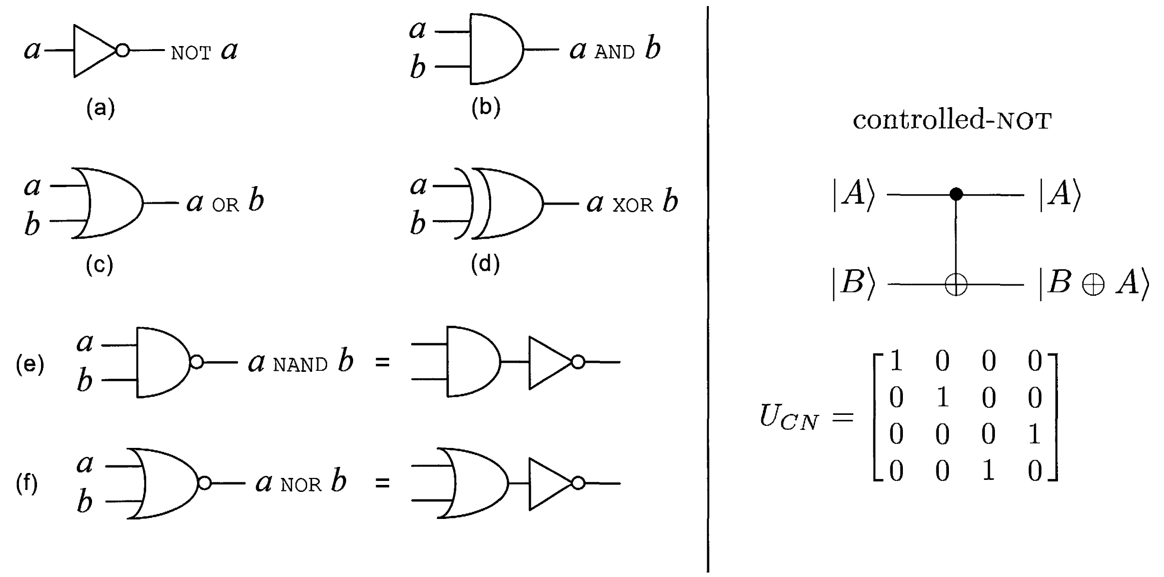


Figure 1.6. On the left are some standard single and multiple bit gates, while on the right is the prototypical multiple qubit gate, the controlled-NOT. The matrix representation of the controlled-NOT, U_{CN}, is written with respect to the amplitudes for $|00\rangle$, $|01\rangle$, $|10\rangle$, and $|11\rangle$, in that order.

qubit. The action of the gate may be described as follows. If the control qubit is set to 0, then the target qubit is left alone. If the control qubit is set to 1, then the target qubit is flipped. In equations:

$$|00\rangle \rightarrow |00\rangle; \; |01\rangle \rightarrow |01\rangle; \; |10\rangle \rightarrow |11\rangle; \; |11\rangle \rightarrow |10\rangle. \tag{1.18}$$

Another way of describing the CNOT is as a generalization of the classical XOR gate, since the action of the gate may be summarized as $|A, B\rangle \rightarrow |A, B \oplus A\rangle$, where $\oplus$ is addition modulo two, which is exactly what the XOR gate does. That is, the control qubit and the target qubit are XORed and stored in the target qubit.

Yet another way of describing the action of the CNOT is to give a matrix representation, as shown in the bottom right of Figure 1.6. You can easily verify that the first column of U_{CN} describes the transformation that occurs to $|00\rangle$, and similarly for the other computational basis states, $|01\rangle$, $|10\rangle$, and $|11\rangle$. As for the single qubit case, the requirement that probability be conserved is expressed in the fact that U_{CN} is a *unitary matrix*, that is, $U_{CN}^\dagger U_{CN} = I$.

We noticed that the CNOT can be regarded as a type of generalized-XOR gate. Can other classical gates such as the NAND or the regular XOR gate be understood as unitary gates in a sense similar to the way the quantum NOT gate represents the classical NOT gate? It turns out that this is not possible. The reason is because the XOR and NAND gates are essentially *irreversible* or *non-invertible*. For example, given the output $A \oplus B$ from an XOR gate, it is not possible to determine what the inputs A and B were; there is an irretrievable *loss of information* associated with the irreversible action of the XOR gate. On the other hand, unitary quantum gates are *always* invertible, since the inverse of a unitary matrix is also a unitary matrix, and thus a quantum gate can always be inverted by another quantum gate. Understanding how to do classical logic in this *reversible* or *invertible* sense will be a crucial step in understanding how to harness the power of quantum mechanics for computation. We'll explain the basic idea of how to do reversible computation in Section 1.4.1.

Of course, there are many interesting quantum gates other than the controlled-NOT. However, in a sense the controlled-NOT and single qubit gates are the prototypes for *all* other gates because of the following remarkable *universality* result: *Any multiple qubit logic gate may be composed from* CNOT *and single qubit gates.* The proof is given in Section 4.5, and is the quantum parallel of the universality of the NAND gate.

1.3.3 Measurements in bases other than the computational basis

We've described quantum measurements of a single qubit in the state $\alpha|0\rangle + \beta|1\rangle$ as yielding the result 0 or 1 and leaving the qubit in the corresponding state $|0\rangle$ or $|1\rangle$, with respective probabilities $|\alpha|^2$ and $|\beta|^2$. In fact, quantum mechanics allows somewhat more versatility in the class of measurements that may be performed, although certainly nowhere near enough to recover α and β from a single measurement!

Note that the states $|0\rangle$ and $|1\rangle$ represent just one of many possible choices of basis states for a qubit. Another possible choice is the set $|+\rangle \equiv (|0\rangle + |1\rangle)/\sqrt{2}$ and $|-\rangle \equiv (|0\rangle - |1\rangle)/\sqrt{2}$. An arbitrary state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ can be re-expressed in terms of the states $|+\rangle$ and $|-\rangle$:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle = \alpha\frac{|+\rangle + |-\rangle}{\sqrt{2}} + \beta\frac{|+\rangle - |-\rangle}{\sqrt{2}} = \frac{\alpha+\beta}{\sqrt{2}}|+\rangle + \frac{\alpha-\beta}{\sqrt{2}}|-\rangle. \quad (1.19)$$

It turns out that it is possible to treat the $|+\rangle$ and $|-\rangle$ states as though they were the computational basis states, and measure with respect to this new basis. Naturally, measuring with respect to the $|+\rangle$, $|-\rangle$ basis results in the result '+' with probability $|\alpha+\beta|^2/2$ and the result '−' with probability $|\alpha - \beta|^2/2$, with corresponding post-measurement states $|+\rangle$ and $|-\rangle$, respectively.

More generally, given any basis states $|a\rangle$ and $|b\rangle$ for a qubit, it is possible to express an arbitrary state as a linear combination $\alpha|a\rangle + \beta|b\rangle$ of those states. Furthermore, provided the states are *orthonormal*, it is possible to *perform a measurement with respect to the* $|a\rangle, |b\rangle$ *basis*, giving the result a with probability $|\alpha|^2$ and b with probability $|\beta|^2$. The orthonormality constraint is necessary in order that $|\alpha|^2 + |\beta|^2 = 1$ as we expect for probabilities. In an analogous way it is possible in principle to measure a quantum system of many qubits with respect to an arbitrary orthonormal basis. However, just because it is possible in principle does not mean that such a measurement can be done easily, and we return later to the question of how efficiently a measurement in an arbitrary basis can be performed.

There are many reasons for using this extended formalism for quantum measurements, but ultimately the best one is this: the formalism allows us to describe observed experimental results, as we will see in our discussion of the Stern–Gerlach experiment in Section 1.5.1. An even more sophisticated and convenient (but essentially equivalent) formalism for describing quantum measurements is described in the next chapter, in Section 2.2.3.

1.3.4 Quantum circuits

We've already met a few simple quantum circuits. Let's look in a little more detail at the elements of a quantum circuit. A simple quantum circuit containing three quantum gates is shown in Figure 1.7. The circuit is to be read from left-to-right. Each line in the circuit represents a *wire* in the quantum circuit. This wire does not necessarily correspond to a physical wire; it may correspond instead to the passage of time, or perhaps to a physical particle such as a photon – a particle of light – moving from one location to another through space. It is conventional to assume that the state input to the circuit is a computational basis state, usually the state consisting of all $|0\rangle$s. This rule is broken frequently in the literature on quantum computation and quantum information, but it is considered polite to inform the reader when this is the case.

The circuit in Figure 1.7 accomplishes a simple but useful task – it swaps the states of the two qubits. To see that this circuit accomplishes the swap operation, note that the sequence of gates has the following sequence of effects on a computational basis state $|a, b\rangle$,

$$\begin{aligned} |a, b\rangle &\longrightarrow |a, a \oplus b\rangle \\ &\longrightarrow |a \oplus (a \oplus b), a \oplus b\rangle = |b, a \oplus b\rangle \\ &\longrightarrow |b, (a \oplus b) \oplus b\rangle = |b, a\rangle , \end{aligned} \tag{1.20}$$

where all additions are done modulo 2. The effect of the circuit, therefore, is to interchange the state of the two qubits.

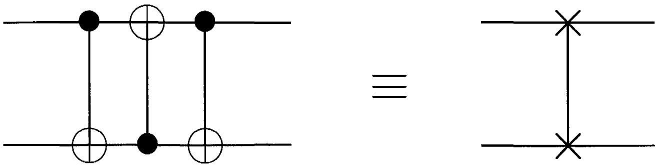


Figure 1.7. Circuit swapping two qubits, and an equivalent schematic symbol notation for this common and useful circuit.

There are a few features allowed in classical circuits that are not usually present in quantum circuits. First of all, we don't allow 'loops', that is, feedback from one part of the quantum circuit to another; we say the circuit is *acyclic*. Second, classical circuits allow wires to be 'joined' together, an operation known as FANIN, with the resulting single wire containing the bitwise OR of the inputs. Obviously this operation is not reversible and therefore not unitary, so we don't allow FANIN in our quantum circuits. Third, the inverse operation, FANOUT, whereby several copies of a bit are produced is also not allowed in quantum circuits. In fact, it turns out that quantum mechanics forbids the copying of a qubit, making the FANOUT operation impossible! We'll see an example of this in the next section when we attempt to design a circuit to copy a qubit.

As we proceed we'll introduce new quantum gates as needed. It's convenient to introduce another convention about quantum circuits at this point. This convention is illustrated in Figure 1.8. Suppose U is *any* unitary matrix acting on some number n of qubits, so U can be regarded as a quantum gate on those qubits. Then we can define a *controlled-U* gate which is a natural extension of the controlled-NOT gate. Such a gate has a single *control qubit*, indicated by the line with the black dot, and n *target qubits*, indicated by the boxed U. If the control qubit is set to 0 then nothing happens to the target qubits. If the control qubit is set to 1 then the gate U is applied to the target qubits. The prototypical example of the controlled-U gate is the controlled-NOT gate, which is a controlled-U gate with $U = X$, as illustrated in Figure 1.9.

Another important operation is measurement, which we represent by a 'meter' symbol,

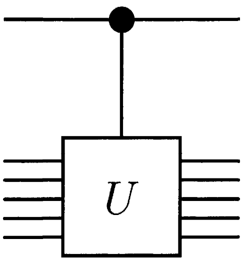


Figure 1.8. Controlled-U gate.

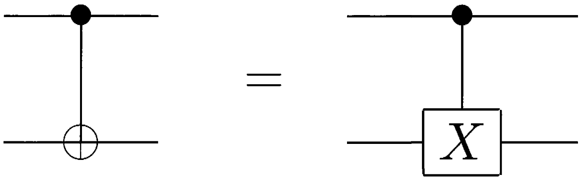


Figure 1.9. Two different representations for the controlled-NOT.

as shown in Figure 1.10. As previously described, this operation converts a single qubit state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ into a probabilistic classical bit M (distinguished from a qubit by drawing it as a double-line wire), which is 0 with probability $|\alpha|^2$, or 1 with probability $|\beta|^2$.

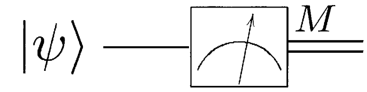


Figure 1.10. Quantum circuit symbol for measurement.

We shall find quantum circuits useful as models of all quantum processes, including but not limited to computation, communication, and even quantum noise. Several simple examples illustrate this below.

1.3.5 Qubit copying circuit?

The CNOT gate is useful for demonstrating one particularly fundamental property of quantum information. Consider the task of copying a classical bit. This may be done using a classical CNOT gate, which takes in the bit to copy (in some unknown state x) and a 'scratchpad' bit initialized to zero, as illustrated in Figure 1.11. The output is two bits, both of which are in the same state x.

Suppose we try to copy a qubit in the unknown state $|\psi\rangle = a\,|0\rangle + b\,|1\rangle$ in the same manner by using a CNOT gate. The input state of the two qubits may be written as

$$\left[a\,|0\rangle + b\,|1\rangle\right]|0\rangle = a\,|00\rangle + b\,|10\rangle, \tag{1.21}$$

The function of CNOT is to negate the second qubit when the first qubit is 1, and thus the output is simply $a\,|00\rangle + b\,|11\rangle$. Have we successfully copied $|\psi\rangle$? That is, have we created the state $|\psi\rangle|\psi\rangle$? In the case where $|\psi\rangle = |0\rangle$ or $|\psi\rangle = |1\rangle$ that is indeed what this circuit does; it is possible to use quantum circuits to copy classical information encoded as a $|0\rangle$ or a $|1\rangle$. However, for a general state $|\psi\rangle$ we see that

$$|\psi\rangle|\psi\rangle = a^2|00\rangle + ab|01\rangle + ab|10\rangle + b^2|11\rangle. \tag{1.22}$$

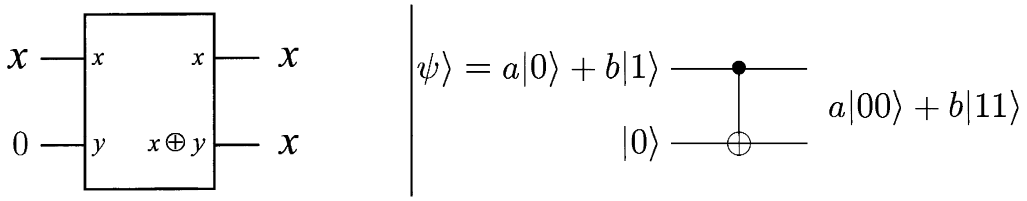


Figure 1.11. Classical and quantum circuits to 'copy' an unknown bit or qubit.

Comparing with $a|00\rangle + b|11\rangle$, we see that unless $ab = 0$ the 'copying circuit' above does *not* copy the quantum state input. In fact, it turns out to be *impossible* to make a copy of an unknown quantum state. This property, that qubits cannot be copied, is known as the *no-cloning* theorem, and it is one of the chief differences between quantum and classical information. The no-cloning theorem is discussed at more length in Box 12.1 on page 532; the proof is very simple, and we encourage you to skip ahead and read the proof now.

There is another way of looking at the failure of the circuit in Figure 1.11, based on the intuition that a qubit somehow contains 'hidden' information not directly accessible to measurement. Consider what happens when we measure one of the qubits of the state $a|00\rangle + b|11\rangle$. As previously described, we obtain either 0 or 1 with probabilities $|a|^2$ and $|b|^2$. However, once one qubit is measured, the state of the other one is completely determined, and no additional information can be gained about a and b. In this sense, the extra hidden information carried in the original qubit $|\psi\rangle$ was lost in the first measurement, and cannot be regained. If, however, the qubit had been copied, then the state of the other qubit should still contain some of that hidden information. Therefore, a copy cannot have been created.

1.3.6 Example: Bell states

Let's consider a slightly more complicated circuit, shown in Figure 1.12, which has a Hadamard gate followed by a CNOT, and transforms the four computational basis states according to the table given. As an explicit example, the Hadamard gate takes the input $|00\rangle$ to $(|0\rangle + |1\rangle)|0\rangle/\sqrt{2}$, and then the CNOT gives the output state $(|00\rangle + |11\rangle)/\sqrt{2}$. Note how this works: first, the Hadamard transform puts the top qubit in a superposition; this then acts as a control input to the CNOT, and the target gets inverted only when the control is 1. The output states

$$|\beta_{00}\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}}; \tag{1.23}$$

$$|\beta_{01}\rangle = \frac{|01\rangle + |10\rangle}{\sqrt{2}}; \tag{1.24}$$

$$|\beta_{10}\rangle = \frac{|00\rangle - |11\rangle}{\sqrt{2}}; \text{ and} \tag{1.25}$$

$$|\beta_{11}\rangle = \frac{|01\rangle - |10\rangle}{\sqrt{2}}, \tag{1.26}$$

are known as the *Bell states*, or sometimes the *EPR states* or *EPR pairs*, after some of the people – Bell, and Einstein, Podolsky, and Rosen – who first pointed out the strange properties of states like these. The mnemonic notation $|\beta_{00}\rangle, |\beta_{01}\rangle, |\beta_{10}\rangle, |\beta_{11}\rangle$ may be understood via the equations

$$|\beta_{xy}\rangle \equiv \frac{|0,y\rangle + (-1)^x|1,\bar{y}\rangle}{\sqrt{2}}, \tag{1.27}$$

where $\bar{y}$ is the negation of y.

In	Out
$\|00\rangle$	$(\|00\rangle + \|11\rangle)/\sqrt{2} \equiv \|\beta_{00}\rangle$
$\|01\rangle$	$(\|01\rangle + \|10\rangle)/\sqrt{2} \equiv \|\beta_{01}\rangle$
$\|10\rangle$	$(\|00\rangle - \|11\rangle)/\sqrt{2} \equiv \|\beta_{10}\rangle$
$\|11\rangle$	$(\|01\rangle - \|10\rangle)/\sqrt{2} \equiv \|\beta_{11}\rangle$

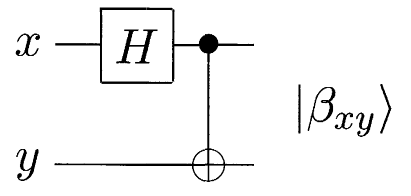


Figure 1.12. Quantum circuit to create Bell states, and its input–ouput quantum 'truth table'.

1.3.7 Example: quantum teleportation

We will now apply the techniques of the last few pages to understand something nontrivial, surprising, and a lot of fun – quantum teleportation! Quantum teleportation is a technique for moving quantum states around, even in the absence of a quantum communications channel linking the sender of the quantum state to the recipient.

Here's how quantum teleportation works. Alice and Bob met long ago but now live far apart. While together they generated an EPR pair, each taking one qubit of the EPR pair when they separated. Many years later, Bob is in hiding, and Alice's mission, should she choose to accept it, is to deliver a qubit $|\psi\rangle$ to Bob. She does not know the state of the qubit, and moreover can only send *classical* information to Bob. Should Alice accept the mission?

Intuitively, things look pretty bad for Alice. She doesn't know the state $|\psi\rangle$ of the qubit she has to send to Bob, and the laws of quantum mechanics prevent her from determining the state when she only has a single copy of $|\psi\rangle$ in her possession. What's worse, even if she did know the state $|\psi\rangle$, describing it precisely takes an infinite amount of classical information since $|\psi\rangle$ takes values in a *continuous* space. So even if she did know $|\psi\rangle$, it would take forever for Alice to describe the state to Bob. It's not looking good for Alice. Fortunately for Alice, quantum teleportation is a way of utilizing the entangled EPR pair in order to send $|\psi\rangle$ to Bob, with only a small overhead of classical communication.

In outline, the steps of the solution are as follows: Alice interacts the qubit $|\psi\rangle$ with her half of the EPR pair, and then measures the two qubits in her possession, obtaining one of four possible classical results, 00, 01, 10, and 11. She sends this information to Bob. Depending on Alice's classical message, Bob performs one of four operations on his half of the EPR pair. Amazingly, by doing this he can recover the original state $|\psi\rangle$!

The quantum circuit shown in Figure 1.13 gives a more precise description of quantum teleportation. The state to be teleported is $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, where α and β are unknown amplitudes. The state input into the circuit $|\psi_0\rangle$ is

$$|\psi_0\rangle = |\psi\rangle|\beta_{00}\rangle \tag{1.28}$$

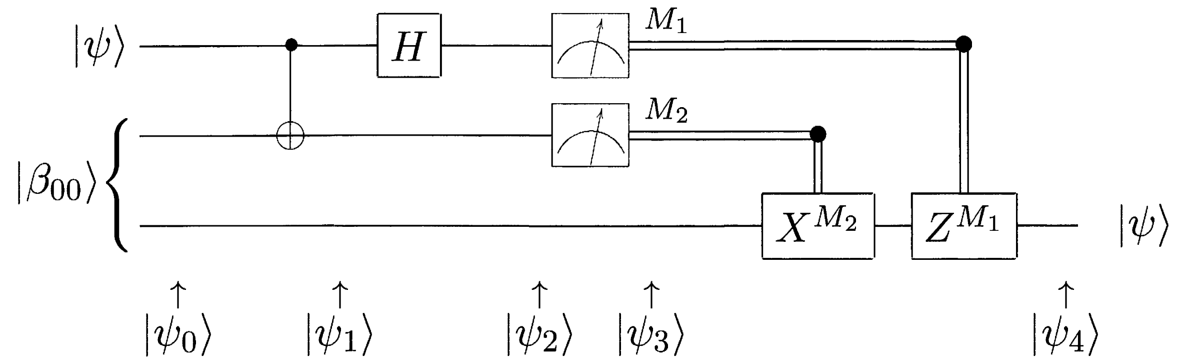


Figure 1.13. Quantum circuit for teleporting a qubit. The two top lines represent Alice's system, while the bottom line is Bob's system. The meters represent measurement, and the double lines coming out of them carry classical bits (recall that single lines denote qubits).

$$= \frac{1}{\sqrt{2}} \Big[\alpha|0\rangle(|00\rangle + |11\rangle) + \beta|1\rangle(|00\rangle + |11\rangle)\Big] , \tag{1.29}$$

where we use the convention that the first two qubits (on the left) belong to Alice, and the third qubit to Bob. As we explained previously, Alice's second qubit and Bob's qubit start out in an EPR state. Alice sends her qubits through a CNOT gate, obtaining

$$|\psi_1\rangle = \frac{1}{\sqrt{2}} \Big[\alpha|0\rangle(|00\rangle + |11\rangle) + \beta|1\rangle(|10\rangle + |01\rangle)\Big] . \tag{1.30}$$

She then sends the first qubit through a Hadamard gate, obtaining

$$|\psi_2\rangle = \frac{1}{2} \Big[\alpha(|0\rangle + |1\rangle)(|00\rangle + |11\rangle) + \beta(|0\rangle - |1\rangle)(|10\rangle + |01\rangle)\Big] . \tag{1.31}$$

This state may be re-written in the following way, simply by regrouping terms:

$$\begin{aligned} |\psi_2\rangle = \frac{1}{2} \Big[&|00\rangle \left(\alpha|0\rangle + \beta|1\rangle\right) + |01\rangle \left(\alpha|1\rangle + \beta|0\rangle\right) \\ &+ |10\rangle \left(\alpha|0\rangle - \beta|1\rangle\right) + |11\rangle \left(\alpha|1\rangle - \beta|0\rangle\right)\Big] . \end{aligned} \tag{1.32}$$

This expression naturally breaks down into four terms. The first term has Alice's qubits in the state $|00\rangle$, and Bob's qubit in the state $\alpha|0\rangle + \beta|1\rangle$ – which is the original state $|\psi\rangle$. If Alice performs a measurement and obtains the result 00 then Bob's system will be in the state $|\psi\rangle$. Similarly, from the previous expression we can read off Bob's post-measurement state, given the result of Alice's measurement:

$$00 \longmapsto |\psi_3(00)\rangle \equiv \Big[\alpha|0\rangle + \beta|1\rangle\Big] \tag{1.33}$$

$$01 \longmapsto |\psi_3(01)\rangle \equiv \Big[\alpha|1\rangle + \beta|0\rangle\Big] \tag{1.34}$$

$$10 \longmapsto |\psi_3(10)\rangle \equiv \Big[\alpha|0\rangle - \beta|1\rangle\Big] \tag{1.35}$$

$$11 \longmapsto |\psi_3(11)\rangle \equiv \Big[\alpha|1\rangle - \beta|0\rangle\Big] . \tag{1.36}$$

Depending on Alice's measurement outcome, Bob's qubit will end up in one of these four possible states. Of course, to know which state it is in, Bob must be told the result of Alice's measurement – we will show later that it is this fact which prevents teleportation from being used to transmit information faster than light. Once Bob has learned the measurement outcome, Bob can 'fix up' his state, recovering $|\psi\rangle$, by applying the appropriate quantum gate. For example, in the case where the measurement yields 00, Bob doesn't need to do anything. If the measurement is 01 then Bob can fix up his state by applying the X gate. If the measurement is 10 then Bob can fix up his state by applying the Z gate. If the measurement is 11 then Bob can fix up his state by applying first an X and then a Z gate. Summing up, Bob needs to apply the transformation $Z^{M_1}X^{M_2}$ (note how time goes from left to right in circuit diagrams, but in matrix products terms on the *right* happen *first*) to his qubit, and he will recover the state $|\psi\rangle$.

There are many interesting features of teleportation, some of which we shall return to later in the book. For now we content ourselves with commenting on a couple of aspects. First, doesn't teleportation allow one to transmit quantum states faster than light? This would be rather peculiar, because the theory of relativity implies that faster than light information transfer could be used to send information backwards in time. Fortunately, quantum teleportation does not enable faster than light communication, because to complete the teleportation Alice must transmit her measurement result to Bob over a classical communications channel. We will show in Section 2.4.3 that without this classical communication, teleportation does not convey *any* information at all. The classical channel is limited by the speed of light, so it follows that quantum teleportation cannot be accomplished faster than the speed of light, resolving the apparent paradox.

A second puzzle about teleportation is that it appears to create a copy of the quantum state being teleported, in apparent violation of the no-cloning theorem discussed in Section 1.3.5. This violation is only illusory since after the teleportation process only the target qubit is left in the state $|\psi\rangle$, and the original data qubit ends up in one of the computational basis states $|0\rangle$ or $|1\rangle$, depending upon the measurement result on the first qubit.

What can we learn from quantum teleportation? Quite a lot! It's much more than just a neat trick one can do with quantum states. Quantum teleportation emphasizes the interchangeability of *different* resources in quantum mechanics, showing that one shared EPR pair together with two classical bits of communication is a resource at least the equal of one qubit of communication. Quantum computation and quantum information has revealed a plethora of methods for interchanging resources, many built upon quantum teleportation. In particular, in Chapter 10 we explain how teleportation can be used to build quantum gates which are resistant to the effects of noise, and in Chapter 12 we show that teleportation is intimately connected with the properties of quantum error-correcting codes. Despite these connections with other subjects, it is fair to say that we are only beginning to understand *why* it is that quantum teleportation is possible in quantum mechanics; in later chapters we endeavor to explain some of the insights that make such an understanding possible.

1.4 Quantum algorithms

What class of computations can be performed using quantum circuits? How does that class compare with the computations which can be performed using classical logical circuits? Can we find a task which a quantum computer may perform better than a classical computer? In this section we investigate these questions, explaining how to perform classical computations on quantum computers, giving some examples of problems for which quantum computers offer an advantage over classical computers, and summarizing the known quantum algorithms.

1.4.1 Classical computations on a quantum computer

Can we simulate a classical logic circuit using a quantum circuit? Not surprisingly, the answer to this question turns out to be yes. It would be very surprising if this were not the case, as physicists believe that all aspects of the world around us, including classical logic circuits, can ultimately be explained using quantum mechanics. As pointed out earlier, the reason quantum circuits cannot be used to directly simulate classical circuits is because unitary quantum logic gates are inherently *reversible*, whereas many classical logic gates such as the NAND gate are inherently irreversible.

Any classical circuit can be replaced by an equivalent circuit containing only *reversible* elements, by making use of a reversible gate known as the *Toffoli gate*. The Toffoli gate has three input bits and three output bits, as illustrated in Figure 1.14. Two of the bits are *control bits* that are unaffected by the action of the Toffoli gate. The third bit is a *target bit* that is flipped if both control bits are set to 1, and otherwise is left alone. Note that applying the Toffoli gate twice to a set of bits has the effect $(a, b, c) \rightarrow (a, b, c \oplus ab) \rightarrow (a, b, c)$, and thus the Toffoli gate is a reversible gate, since it has an inverse – itself.

Inputs			Outputs		
a	b	c	a'	b'	c'
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

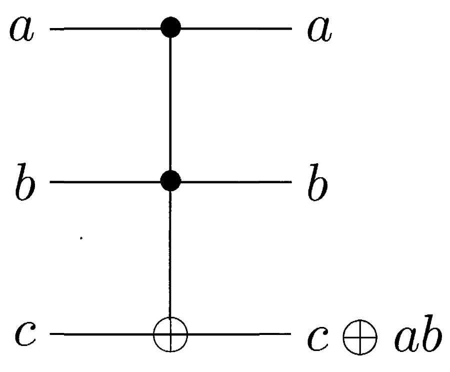


Figure 1.14. Truth table for the Toffoli gate, and its circuit representation.

The Toffoli gate can be used to simulate NAND gates, as shown in Figure 1.15, and can also be used to do FANOUT, as shown in Figure 1.16. With these two operations it becomes possible to simulate all other elements in a classical circuit, and thus an arbitrary classical circuit can be simulated by an equivalent reversible circuit.

The Toffoli gate has been described as a classical gate, but it can also be implemented as a quantum logic gate. By definition, the quantum logic implementation of the Toffoli gate simply permutes computational basis states in the same way as the classical Toffoli gate. For example, the quantum Toffoli gate acting on the state $|110\rangle$ flips the third qubit because the first two are set, resulting in the state $|111\rangle$. It is tedious but not difficult to write this transformation out as an 8 by 8 matrix, U, and verify explicitly that U is a unitary matrix, and thus the Toffoli gate is a legitimate quantum gate. The quantum Toffoli gate can be used to simulate irreversible classical logic gates, just as the classical

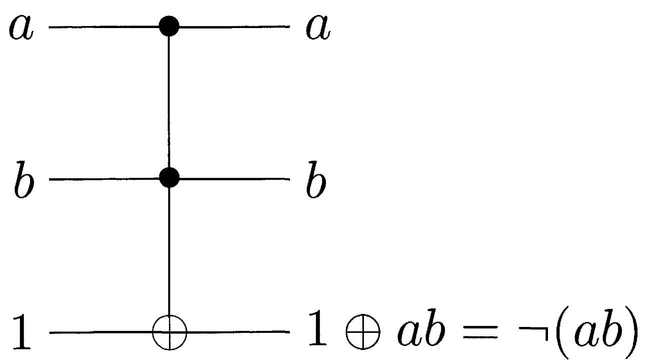


Figure 1.15. Classical circuit implementing a NAND gate using a Toffoli gate. The top two bits represent the input to the NAND, while the third bit is prepared in the standard state 1, sometimes known as an *ancilla* state. The output from the NAND is on the third bit.

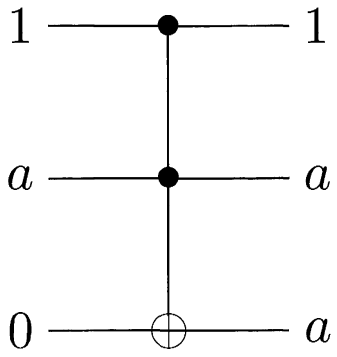


Figure 1.16. FANOUT with the Toffoli gate, with the second bit being the input to the FANOUT (and the other two bits standard ancilla states), and the output from FANOUT appearing on the second and third bits.

Toffoli gate was, and ensures that quantum computers are capable of performing any computation which a classical (deterministic) computer may do.

What if the classical computer is non-deterministic, that is, has the ability to generate random bits to be used in the computation? Not surprisingly, it is easy for a quantum computer to simulate this. To perform such a simulation it turns out to be sufficient to produce random fair coin tosses, which can be done by preparing a qubit in the state $|0\rangle$, sending it through a Hadamard gate to produce $(|0\rangle + |1\rangle)/\sqrt{2}$, and then measuring the state. The result will be $|0\rangle$ or $|1\rangle$ with 50/50 probability. This provides a quantum computer with the ability to efficiently simulate a non-deterministic classical computer.

Of course, if the ability to simulate classical computers were the only feature of quantum computers there would be little point in going to all the trouble of exploiting quantum effects! The advantage of quantum computing is that much more powerful functions may be computed using qubits and quantum gates. In the next few sections we explain how to do this, culminating in the Deutsch–Jozsa algorithm, our first example of a quantum algorithm able to solve a problem faster than any classical algorithm.

1.4.2 Quantum parallelism

Quantum parallelism is a fundamental feature of many quantum algorithms. Heuristically, and at the risk of over-simplifying, quantum parallelism allows quantum computers to evaluate a function $f(x)$ for many *different* values of x simultaneously. In this section we explain how quantum parallelism works, and some of its limitations.

Suppose $f(x) : \{0,1\} \rightarrow \{0,1\}$ is a function with a one-bit domain and range. A convenient way of computing this function on a quantum computer is to consider a two qubit quantum computer which starts in the state $|x, y\rangle$. With an appropriate sequence of logic gates it is possible to transform this state into $|x, y \oplus f(x)\rangle$, where $\oplus$ indicates addition modulo 2; the first register is called the 'data' register, and the second register the 'target' register. We give the transformation defined by the map $|x, y\rangle \rightarrow |x, y \oplus f(x)\rangle$ a name, U_f, and note that it is easily shown to be unitary. If $y = 0$, then the final state of the second qubit is just the value $f(x)$. (In Section 3.2.5 we show that given a classical circuit for computing f there is a quantum circuit of comparable efficiency which computes the transformation U_f on a quantum computer. For our purposes it can be considered to be a black box.)

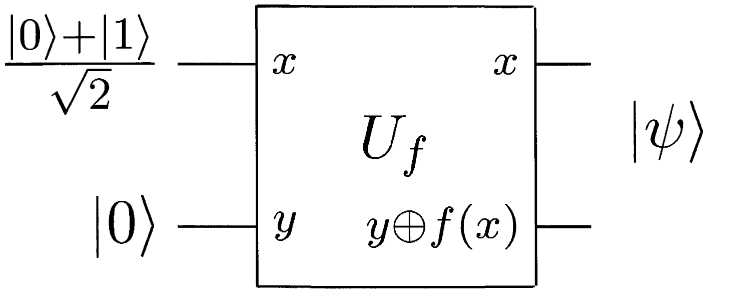


Figure 1.17. Quantum circuit for evaluating $f(0)$ and $f(1)$ simultaneously. U_f is the quantum circuit which takes inputs like $|x, y\rangle$ to $|x, y \oplus f(x)\rangle$.

Consider the circuit shown in Figure 1.17, which applies U_f to an input not in the computational basis. Instead, the data register is prepared in the superposition $(|0\rangle + |1\rangle)/\sqrt{2}$, which can be created with a Hadamard gate acting on $|0\rangle$. Then we apply U_f, resulting in the state:

$$\frac{|0, f(0)\rangle + |1, f(1)\rangle}{\sqrt{2}}. \tag{1.37}$$

This is a remarkable state! The different terms contain information about both $f(0)$ and $f(1)$; it is almost as if we have evaluated $f(x)$ for two values of x simultaneously, a feature known as 'quantum parallelism'. Unlike classical parallelism, where multiple circuits each built to compute $f(x)$ are executed simultaneously, here a *single* $f(x)$ circuit is employed to evaluate the function for multiple values of x simultaneously, by exploiting the ability of a quantum computer to be in superpositions of different states.

This procedure can easily be generalized to functions on an arbitrary number of bits, by using a general operation known as the *Hadamard transform*, or sometimes the *Walsh–Hadamard transform*. This operation is just n Hadamard gates acting in parallel on n qubits. For example, shown in Figure 1.18 is the case $n = 2$ with qubits initially prepared as $|0\rangle$, which gives

$$\left(\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right)\left(\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right) = \frac{|00\rangle + |01\rangle + |10\rangle + |11\rangle}{2} \tag{1.38}$$

as output. We write $H^{\otimes 2}$ to denote the parallel action of two Hadamard gates, and read '$\otimes$' as 'tensor'. More generally, the result of performing the Hadamard transform on n

qubits initially in the all $|0\rangle$ state is

$$\frac{1}{\sqrt{2^n}}\sum_x |x\rangle, \tag{1.39}$$

where the sum is over all possible values of x, and we write $H^{\otimes n}$ to denote this action. That is, the Hadamard transform produces an equal superposition of all computational basis states. Moreover, it does this extremely efficiently, producing a superposition of 2^n states using just n gates.

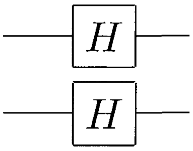


Figure 1.18. The Hadamard transform $H^{\otimes 2}$ on two qubits.

Quantum parallel evaluation of a function with an n bit input x and 1 bit output, $f(x)$, can thus be performed in the following manner. Prepare the $n+1$ qubit state $|0\rangle^{\otimes n}|0\rangle$, then apply the Hadamard transform to the first n qubits, followed by the quantum circuit implementing U_f. This produces the state

$$\frac{1}{\sqrt{2^n}}\sum_x |x\rangle|f(x)\rangle. \tag{1.40}$$

In some sense, quantum parallelism enables all possible values of the function f to be evaluated simultaneously, even though we apparently only evaluated f once. However, this parallelism is *not* immediately useful. In our single qubit example, measurement of the state gives only *either* $|0, f(0)\rangle$ *or* $|1, f(1)\rangle$! Similarly, in the general case, measurement of the state $\sum_x |x, f(x)\rangle$ would give only $f(x)$ for a single value of x. Of course, a classical computer can do this easily! Quantum computation requires something more than just quantum parallelism to be useful; it requires the ability to *extract* information about more than one value of $f(x)$ from superposition states like $\sum_x |x, f(x)\rangle$. Over the next two sections we investigate examples of how this may be done.

1.4.3 Deutsch's algorithm

A simple modification of the circuit in Figure 1.17 demonstrates how quantum circuits can outperform classical ones by implementing *Deutsch's algorithm* (we actually present a simplified and improved version of the original algorithm; see 'History and further reading' at the end of the chapter). Deutsch's algorithm combines quantum parallelism with a property of quantum mechanics known as *interference*. As before, let us use the Hadamard gate to prepare the first qubit as the superposition $(|0\rangle + |1\rangle)/\sqrt{2}$, but now let us prepare the second qubit y as the superposition $(|0\rangle - |1\rangle)/\sqrt{2}$, using a Hadamard gate applied to the state $|1\rangle$. Let us follow the states along to see what happens in this circuit, shown in Figure 1.19.

The input state

$$|\psi_0\rangle = |01\rangle \tag{1.41}$$

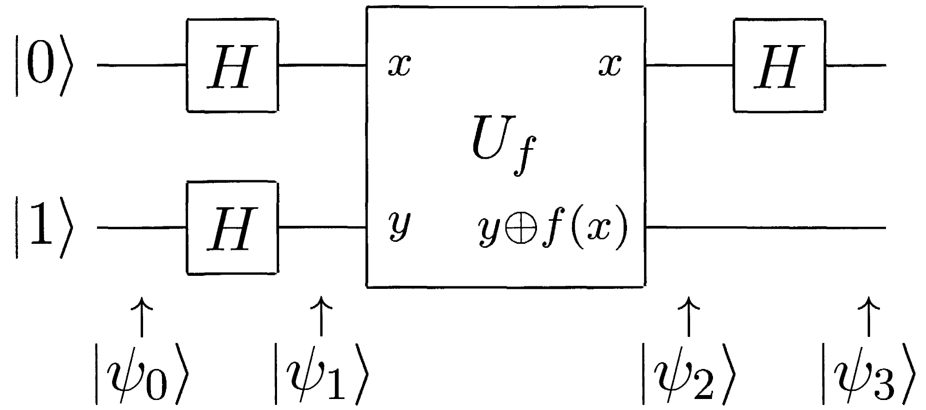


Figure 1.19. Quantum circuit implementing Deutsch's algorithm.

is sent through two Hadamard gates to give

$$|\psi_1\rangle = \left[\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right]\left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right]. \tag{1.42}$$

A little thought shows that if we apply U_f to the state $|x\rangle(|0\rangle - |1\rangle)/\sqrt{2}$ then we obtain the state $(-1)^{f(x)}|x\rangle(|0\rangle - |1\rangle)/\sqrt{2}$. Applying U_f to $|\psi_1\rangle$ therefore leaves us with one of two possibilities:

$$|\psi_2\rangle = \begin{cases} \pm\left[\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right]\left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right] & \text{if } f(0) = f(1) \\ \pm\left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right]\left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right] & \text{if } f(0) \neq f(1). \end{cases} \tag{1.43}$$

The final Hadamard gate on the first qubit thus gives us

$$|\psi_3\rangle = \begin{cases} \pm|0\rangle\left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right] & \text{if } f(0) = f(1) \\ \pm|1\rangle\left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right] & \text{if } f(0) \neq f(1). \end{cases} \tag{1.44}$$

Realizing that $f(0) \oplus f(1)$ is 0 if $f(0) = f(1)$ and 1 otherwise, we can rewrite this result concisely as

$$|\psi_3\rangle = \pm|f(0) \oplus f(1)\rangle\left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right], \tag{1.45}$$

so by measuring the first qubit we may determine $f(0) \oplus f(1)$. This is very interesting indeed: the quantum circuit has given us the ability to determine a *global property* of $f(x)$, namely $f(0) \oplus f(1)$, using only *one* evaluation of $f(x)$! This is faster than is possible with a classical apparatus, which would require at least two evaluations.

This example highlights the difference between quantum parallelism and classical randomized algorithms. Naively, one might think that the state $|0\rangle|f(0)\rangle + |1\rangle|f(1)\rangle$ corresponds rather closely to a probabilistic classical computer that evaluates $f(0)$ with probability one-half, or $f(1)$ with probability one-half. The difference is that in a classical computer these two alternatives forever exclude one another; in a quantum computer it is possible for the two alternatives to *interfere* with one another to yield some global property of the function f, by using something like the Hadamard gate to recombine the different alternatives, as was done in Deutsch's algorithm. The essence of the design of many quantum algorithms is that a clever choice of function and final transformation allows efficient determination of useful global information about the function – information which cannot be attained quickly on a classical computer.

1.4.4 The Deutsch–Jozsa algorithm

Deutsch's algorithm is a simple case of a more general quantum algorithm, which we shall refer to as the Deutsch–Jozsa algorithm. The application, known as *Deutsch's problem*, may be described as the following game. Alice, in Amsterdam, selects a number x from 0 to $2^n - 1$, and mails it in a letter to Bob, in Boston. Bob calculates some function $f(x)$ and replies with the result, which is either 0 or 1. Now, Bob has promised to use a function f which is of one of two kinds; either $f(x)$ is *constant* for all values of x, or else $f(x)$ is *balanced*, that is, equal to 1 for exactly half of all the possible x, and 0 for the other half. Alice's goal is to determine with certainty whether Bob has chosen a constant or a balanced function, corresponding with him as little as possible. How fast can she succeed?

In the classical case, Alice may only send Bob one value of x in each letter. At worst, Alice will need to query Bob at least $2^n/2+1$ times, since she may receive $2^n/2$ 0s before finally getting a 1, telling her that Bob's function is balanced. The best deterministic classical algorithm she can use therefore requires $2^n/2 + 1$ queries. Note that in each letter, Alice sends Bob n bits of information. Furthermore, in this example, physical distance is being used to artificially elevate the cost of calculating $f(x)$, but this is not needed in the general problem, where $f(x)$ may be inherently difficult to calculate.

If Bob and Alice were able to exchange qubits, instead of just classical bits, and if Bob agreed to calculate $f(x)$ using a unitary transform U_f, then Alice could achieve her goal in just *one* correspondence with Bob, using the following algorithm.

Analogously to Deutsch's algorithm, Alice has an n qubit register to store her query in, and a single qubit register which she will give to Bob, to store the answer in. She begins by preparing both her query and answer registers in a superposition state. Bob will evaluate $f(x)$ using quantum parallelism and leave the result in the answer register. Alice then interferes states in the superposition using a Hadamard transform on the query register, and finishes by performing a suitable measurement to determine whether f was constant or balanced.

The specific steps of the algorithm are depicted in Figure 1.20. Let us follow the states through this circuit. The input state

$$|\psi_0\rangle = |0\rangle^{\otimes n}|1\rangle \tag{1.46}$$

is similar to that of Equation (1.41), but here the query register describes the state of n qubits all prepared in the $|0\rangle$ state. After the Hadamard transform on the query register and the Hadamard gate on the answer register we have

$$|\psi_1\rangle = \sum_{x\in\{0,1\}^n} \frac{|x\rangle}{\sqrt{2^n}} \left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right]. \tag{1.47}$$

The query register is now a superposition of all values, and the answer register is in an

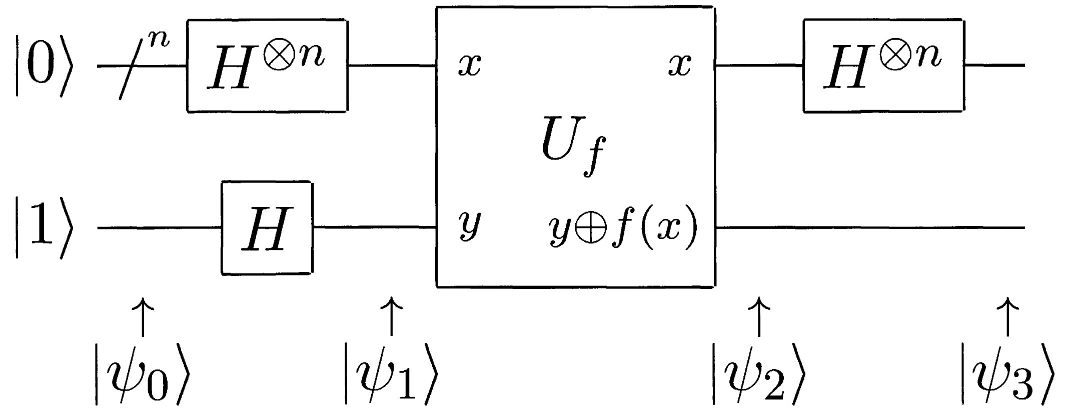


Figure 1.20. Quantum circuit implementing the general Deutsch–Jozsa algorithm. The wire with a '/' through it represents a set of n qubits, similar to the common engineering notation.

evenly weighted superposition of 0 and 1. Next, the function f is evaluated (by Bob) using $U_f : |x, y\rangle \rightarrow |x, y \oplus f(x)\rangle$, giving

$$|\psi_2\rangle = \sum_x \frac{(-1)^{f(x)}|x\rangle}{\sqrt{2^n}} \left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right]. \tag{1.48}$$

Alice now has a set of qubits in which the result of Bob's function evaluation is stored in the amplitude of the qubit superposition state. She now interferes terms in the superposition using a Hadamard transform on the query register. To determine the result of the Hadamard transform it helps to first calculate the effect of the Hadamard transform on a state $|x\rangle$. By checking the cases $x = 0$ and $x = 1$ separately we see that for a single qubit $H|x\rangle = \sum_z (-1)^{xz}|z\rangle/\sqrt{2}$. Thus

$$H^{\otimes n}|x_1, \ldots, x_n\rangle = \frac{\sum_{z_1,\ldots,z_n}(-1)^{x_1z_1+\cdots+x_nz_n}|z_1, \ldots, z_n\rangle}{\sqrt{2^n}}. \tag{1.49}$$

This can be summarized more succinctly in the very useful equation

$$H^{\otimes n}|x\rangle = \frac{\sum_z(-1)^{x\cdot z}|z\rangle}{\sqrt{2^n}}, \tag{1.50}$$

where $x \cdot z$ is the bitwise inner product of x and z, modulo 2. Using this equation and (1.48) we can now evaluate $|\psi_3\rangle$,

$$|\psi_3\rangle = \sum_z \sum_x \frac{(-1)^{x\cdot z+f(x)}|z\rangle}{2^n} \left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right]. \tag{1.51}$$

Alice now observes the query register. Note that the amplitude for the state $|0\rangle^{\otimes n}$ is $\sum_x(-1)^{f(x)}/2^n$. Let's look at the two possible cases – f constant and f balanced – to discern what happens. In the case where f is constant the amplitude for $|0\rangle^{\otimes n}$ is +1 or −1, depending on the constant value $f(x)$ takes. Because $|\psi_3\rangle$ is of unit length it follows that all the other amplitudes must be zero, and an observation will yield 0s for all qubits in the query register. If f is balanced then the positive and negative contributions to the amplitude for $|0\rangle^{\otimes n}$ cancel, leaving an amplitude of zero, and a measurement must yield a result other than 0 on at least one qubit in the query register. Summarizing, if Alice measures all 0s then the function is constant; otherwise the function is balanced. The Deutsch–Jozsa algorithm is summarized below.

Algorithm: Deutsch–Jozsa

Inputs: (1) A black box U_f which performs the transformation $|x\rangle|y\rangle \rightarrow |x\rangle|y \oplus f(x)\rangle$, for $x \in \{0, \ldots, 2^n - 1\}$ and $f(x) \in \{0, 1\}$. It is promised that $f(x)$ is either *constant* for all values of x, or else $f(x)$ is *balanced*, that is, equal to 1 for exactly half of all the possible x, and 0 for the other half.

Outputs: 0 if and only if f is constant.

Runtime: One evaluation of U_f. Always succeeds.

Procedure:

1.	$\|0\rangle^{\otimes n}\|1\rangle$	initialize state
2.	$\rightarrow \frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n-1}\|x\rangle\left[\frac{\|0\rangle-\|1\rangle}{\sqrt{2}}\right]$	create superposition using Hadamard gates
3.	$\rightarrow \sum_x(-1)^{f(x)}\|x\rangle\left[\frac{\|0\rangle-\|1\rangle}{\sqrt{2}}\right]$	calculate function f using U_f
4.	$\rightarrow \sum_z\sum_x\frac{(-1)^{x\cdot z+f(x)}\|z\rangle}{\sqrt{2^n}}\left[\frac{\|0\rangle-\|1\rangle}{\sqrt{2}}\right]$	perform Hadamard transform
5.	$\rightarrow z$	measure to obtain final output z

We've shown that a quantum computer can solve Deutsch's problem with one evaluation of the function f compared to the classical requirement for $2^n/2 + 1$ evaluations. This appears impressive, but there are several important caveats. First, Deutsch's problem is not an especially important problem; it has no known applications. Second, the comparison between classical and quantum algorithms is in some ways an apples and oranges comparison, as the method for evaluating the function is quite different in the two cases. Third, if Alice is allowed to use a probabilistic classical computer, then by asking Bob to evaluate $f(x)$ for a few randomly chosen x she can very quickly determine with high probability whether f is constant or balanced. This probabilistic scenario is perhaps more realistic than the deterministic scenario we have been considering. Despite these caveats, the Deutsch–Jozsa algorithm contains the seeds for more impressive quantum algorithms, and it is enlightening to attempt to understand the principles behind its operation.

Exercise 1.1: (Probabilistic classical algorithm) Suppose that the problem is not to distinguish between the constant and balanced functions *with certainty*, but rather, with some probability of error $\epsilon < 1/2$. What is the performance of the best classical algorithm for this problem?

1.4.5 Quantum algorithms summarized

The Deutsch–Jozsa algorithm suggests that quantum computers may be capable of solving some computational problems much more efficiently than classical computers. Unfortunately, the problem it solves is of little practical interest. Are there more interesting problems whose solution may be obtained more efficiently using quantum algorithms? What are the principles underlying such algorithms? What are the ultimate limits of a quantum computer's computational power?

Broadly speaking, there are three classes of quantum algorithms which provide an advantage over known classical algorithms. First, there is the class of algorithms based upon quantum versions of the Fourier transform, a tool which is also widely used in classical algorithms. The Deutsch–Jozsa algorithm is an example of this type of algorithm, as are Shor's algorithms for factoring and discrete logarithm. The second class of algorithms is quantum search algorithms. The third class of algorithms is quantum simulation, whereby a quantum computer is used to simulate a quantum system. We now briefly describe each of these classes of algorithms, and then summarize what is known or suspected about the computational power of quantum computers.

Quantum algorithms based upon the Fourier transform

The discrete Fourier transform is usually described as transforming a set $x_0, \ldots, x_{N-1}$ of N complex numbers into a set of complex numbers $y_0, \ldots, y_{N-1}$ defined by

$$y_k \equiv \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} e^{2\pi i jk/N} x_j \,. \tag{1.52}$$

Of course, this transformation has an enormous number of applications in many branches of science; the Fourier transformed version of a problem is often easier than the original problem, enabling a solution.

The Fourier transform has proved so useful that a beautiful generalized theory of Fourier transforms has been developed which goes beyond the definition (1.52). This general theory involves some technical ideas from the character theory of finite groups, and we will not attempt to describe it here. What is important is that the Hadamard transform used in the Deutsch–Jozsa algorithm is an example of this generalized class of Fourier transforms. Moreover, many of the other important quantum algorithms also involve some type of Fourier transform.

The most important quantum algorithms known, Shor's fast algorithms for factoring and discrete logarithm, are two examples of algorithms based upon the Fourier transform defined in Equation (1.52). The Equation (1.52) does not appear terribly quantum mechanical in the form we have written it. Imagine, however, that we define a linear transformation U on n qubits by its action on computational basis states $|j\rangle$, where $0 \leq j \leq 2^n - 1$,

$$|j\rangle \longrightarrow \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} e^{2\pi i jk/2^n} |k\rangle \,. \tag{1.53}$$

It can be checked that this transformation is unitary, and in fact can be realized as a quantum circuit. Moreover, if we write out its action on superpositions,

$$\sum_{j=0}^{2^n-1} x_j |j\rangle \longrightarrow \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} \left[\sum_{j=0}^{2^n-1} e^{2\pi i jk/2^n} x_j \right] |k\rangle = \sum_{k=0}^{2^n-1} y_k |k\rangle \,, \tag{1.54}$$

we see that it corresponds to a vector notation for the Fourier transform (1.52) for the case $N = 2^n$.

How quickly can we perform the Fourier transform? Classically, the fast Fourier transform takes roughly $N\log(N) = n2^n$ steps to Fourier transform $N = 2^n$ numbers. On a quantum computer, the Fourier transform can be accomplished using about $\log^2(N) = n^2$ steps, an exponential saving! The quantum circuit to do this is explained in Chapter 5.

This result seems to indicate that quantum computers can be used to very quickly compute the Fourier transform of a vector of 2^n complex numbers, which would be fantastically useful in a wide range of applications. However, that is *not* exactly the case; the Fourier transform is being performed on the information 'hidden' in the amplitudes of the quantum state. This information is not directly accessible to measurement. The catch, of course, is that if the output state is measured, it will collapse each qubit into the state $|0\rangle$ or $|1\rangle$, preventing us from learning the transform result y_k directly. This example speaks to the heart of the conundrum of devising a quantum algorithm. On the one hand, we can perform certain calculations on the 2^n amplitudes associated with n qubits far more efficiently than would be possible on a classical computer. But on the other hand, the results of such a calculation are not available to us if we go about it in a straightforward manner. More cleverness is required in order to harness the power of quantum computation.

Fortunately, it does turn out to be possible to utilize the quantum Fourier transform to efficiently solve several problems that are believed to have no efficient solution on a classical computer. These problems include Deutsch's problem, and Shor's algorithms for discrete logarithm and factoring. This line of thought culminated in Kitaev's discovery of a method to solve the *Abelian stabilizer problem*, and the generalization to the *hidden subgroup problem*,

> Let f be a function from a finitely generated group G to a finite set X such that f is constant on the cosets of a subgroup K, and distinct on each coset. Given a quantum black box for performing the unitary transform $U|g\rangle|h\rangle = |g\rangle|h \oplus f(g)\rangle$, for $g \in G$, $h \in X$, and $\oplus$ an appropriately chosen binary operation on X, find a generating set for K.

The Deutsch–Jozsa algorithm, Shor's algorithms, and related 'exponentially fast' quantum algorithms can all be viewed as special cases of this algorithm. The quantum Fourier transform and its applications are described in Chapter 5.

Quantum search algorithms

A completely different class of algorithms is represented by the quantum search algorithm, whose basic principles were discovered by Grover. The quantum search algorithm solves the following problem: Given a search space of size N, and no prior knowledge about the structure of the information in it, we want to find an element of that search space satisfying a known property. How long does it take to find an element satisfying that property? Classically, this problem requires approximately N operations, but the quantum search algorithm allows it to be solved using approximately $\sqrt{N}$ operations.

The quantum search algorithm offers only a quadratic speedup, as opposed to the more impressive exponential speedup offered by algorithms based on the quantum Fourier transform. However, the quantum search algorithm is still of great interest, since searching heuristics have a wider range of application than the problems solved using the quantum Fourier transform, and adaptations of the quantum search algorithm may have utility for a very wide range of problems. The quantum search algorithm and its applications are described in Chapter 6.

Quantum simulation

Simulating naturally occurring quantum mechanical systems is an obvious candidate for a task at which quantum computers may excel, yet which is believed to be difficult on a classical computer. Classical computers have difficulty simulating general quantum systems for much the same reasons they have difficulty simulating quantum computers – the number of complex numbers needed to describe a quantum system generally grows *exponentially* with the size of the system, rather than linearly, as occurs in classical systems. In general, storing the quantum state of a system with n distinct components takes something like c^n bits of memory on a classical computer, where c is a constant which depends upon details of the system being simulated, and the desired accuracy of the simulation.

By contrast, a quantum computer can perform the simulation using kn qubits, where k is again a constant which depends upon the details of the system being simulated. This allows quantum computers to efficiently perform simulations of quantum mechanical systems that are believed not to be efficiently simulatable on a classical computer. A significant caveat is that even though a quantum computer can simulate many quantum systems far more efficiently than a classical computer, this does not mean that the fast simulation will allow the desired information about the quantum system to be obtained. When measured, a kn qubit simulation will collapse into a definite state, giving only kn bits of information; the c^n bits of 'hidden information' in the wavefunction is not entirely accessible. Thus, a crucial step in making quantum simulations useful is development of systematic means by which desired answers can be efficiently extracted; how to do this is only partially understood.

Despite this caveat, quantum simulation is likely to be an important application of quantum computers. The simulation of quantum systems is an important problem in many fields, notably quantum chemistry, where the computational constraints imposed by classical computers make it difficult to accurately simulate the behavior of even moderately sized molecules, much less the very large molecules that occur in many important biological systems. Obtaining faster and more accurate simulations of such systems may therefore have the welcome effect of enabling advances in other fields in which quantum phenomena are important.

In the future we may discover a physical phenomenon in Nature which cannot be efficiently simulated on a quantum computer. Far from being bad news, this would be wonderful! At the least, it will stimulate us to extend our models of computation to encompass the new phenomenon, and increase the power of our computational models beyond the existing quantum computing model. It also seems likely that very interesting new physical effects will be associated with any such phenomenon!

Another application for quantum simulation is as a general method to obtain insight into other quantum algorithms; for example, in Section 6.2 we explain how the quantum search algorithm can be viewed as the solution to a problem of quantum simulation. By approaching the problem in this fashion it becomes much easier to understand the origin of the quantum search algorithm.

Finally, quantum simulation also gives rise to an interesting and optimistic 'quantum corollary' to Moore's law. Recall that Moore's law states that the power of classical computers will double once every two years or so, for constant cost. However, suppose we are simulating a quantum system on a classical computer, and want to add a single qubit (or a larger system) to the system being simulated. This doubles or more the memory requirements needed for a classical computer to store a description of the state of the quantum system, with a similar or greater cost in the time needed to simulate the dynamics. The quantum corollary to Moore's law follows from this observation, stating that quantum computers are keeping pace with classical computers provided a *single qubit* is added to the quantum computer every two years. This corollary should not be taken too seriously, as the exact nature of the gain, if any, of quantum computation over classical is not yet clear. Nevertheless, this heuristic statement helps convey why we should be interested in quantum computers, and hopeful that they will one day be able to outperform the most powerful classical computers, at least for some applications.

The power of quantum computation

How powerful are quantum computers? What gives them their power? Nobody yet knows the answers to these questions, despite the suspicions fostered by examples such as factoring, which strongly suggest that quantum computers are more powerful than classical computers. It is still possible that quantum computers are no more powerful than classical computers, in the sense that any problem which can be efficiently solved on a quantum computer can also be efficiently solved on a classical computer. On the other hand, it may eventually be proved that quantum computers are much more powerful than classical computers. We now take a brief look at what is known about the power of quantum computation.

Computational complexity theory is the subject of classifying the difficulty of various computational problems, both classical and quantum, and to understand the power of quantum computers we will first examine some general ideas from computational complexity. The most basic idea is that of a *complexity class*. A complexity class can be thought of as a collection of computational problems, all of which share some common feature with respect to the computational resources needed to solve those problems.

Two of the most important complexity classes go by the names **P** and **NP**. Roughly speaking, **P** is the class of computational problems that can be solved quickly on a classical computer. **NP** is the class of problems which have *solutions* which can be quickly checked on a classical computer. To understand the distinction between **P** and **NP**, consider the problem of finding the prime factors of an integer, n. So far as is known there is no fast way of solving this problem on a classical computer, which suggests that the problem is not in **P**. On the other hand, if somebody tells you that some number p is a factor of n, then we can quickly check that this is correct by dividing p into n, so factoring is a problem in **NP**.

It is clear that **P** is a subset of **NP**, since the ability to solve a problem implies the ability to check potential solutions. What is not so clear is whether or not there are problems in **NP** that are not in **P**. Perhaps the most important unsolved problem in theoretical computer science is to determine whether these two classes are different:

$$\mathbf{P} \overset{?}{\neq} \mathbf{NP}. \tag{1.55}$$

Most researchers believe that **NP** contains problems that are not in **P**. In particular, there is an important subclass of the **NP** problems, the **NP**-complete problems, that are of especial importance for two reasons. First, there are thousands of problems, many highly important, that are known to be **NP**-complete. Second, any given **NP**-complete problem is in some sense 'at least as hard' as all other problems in **NP**. More precisely, an algorithm to solve a specific **NP**-complete problem can be adapted to solve any other problem in **NP**, with a small overhead. In particular, if **P** $\neq$ **NP**, then it will follow that no **NP**-complete problem can be efficiently solved on a classical computer.

It is not known whether quantum computers can be used to quickly solve all the problems in **NP**, despite the fact that they can be used to solve some problems – like factoring – which are believed by many people to be in **NP** but not in **P**. (Note that factoring is not known to be **NP**-complete, otherwise we would already know how to efficiently solve all problems in **NP** using quantum computers.) It would certainly be very exciting if it were possible to solve all the problems in **NP** efficiently on a quantum computer. There is a very interesting negative result known in this direction which rules out using a simple variant of quantum parallelism to solve all the problems in **NP**. Specifically, one approach to the problem of solving problems in **NP** on a quantum computer is to try to use some form of quantum parallelism to search in parallel through all the possible solutions to the problem. In Section 6.6 we will show that no approach based upon such a search-based methodology can yield an efficient solution to all the problems in **NP**. While it is disappointing that this approach fails, it does not rule out that some deeper structure exists in the problems in **NP** that will allow them all to be solved quickly using a quantum computer.

P and **NP** are just two of a plethora of complexity classes that have been defined. Another important complexity class is **PSPACE**. Roughly speaking, **PSPACE** consists of those problems which can be solved using resources which are few in spatial size (that is, the computer is 'small'), but not necessarily in time ('long' computations are fine). **PSPACE** is believed to be strictly larger than both **P** and **NP** although, again, this has never been proved. Finally, the complexity class **BPP** is the class of problems that can be solved using randomized algorithms in polynomial time, if a bounded probability of error (say $1/4$) is allowed in the solution to the problem. **BPP** is widely regarded as being, even more so than **P**, the class of problems which should be considered efficiently soluble on a classical computer. We have elected to concentrate here on **P** rather than **BPP** because **P** has been studied in more depth, however many similar ideas and conclusions arise in connection with **BPP**.

What of quantum complexity classes? We can define **BQP** to be the class of all computational problems which can be solved efficiently on a quantum computer, where a bounded probability of error is allowed. (Strictly speaking this makes **BQP** more analogous to the classical complexity class **BPP** than to **P**, however we will ignore this subtlety for the purposes of the present discussion, and treat it as the analogue of **P**.) Exactly where **BQP** fits with respect to **P**, **NP** and **PSPACE** is as yet unknown. What is known is that quantum computers can solve all the problems in **P** efficiently, but that there are no problems outside of **PSPACE** which they can solve efficiently. Therefore, **BQP** lies somewhere between **P** and **PSPACE**, as illustrated in Figure 1.21. An important implication is that if it is proved that quantum computers are strictly more powerful than classical computers, then it will follow that **P** is not equal to **PSPACE**. Proving this latter result has been attempted without success by many computer scientists, suggesting that it may be non-trivial to prove that quantum computers are more powerful than classical computers, despite much evidence in favor of this proposition.

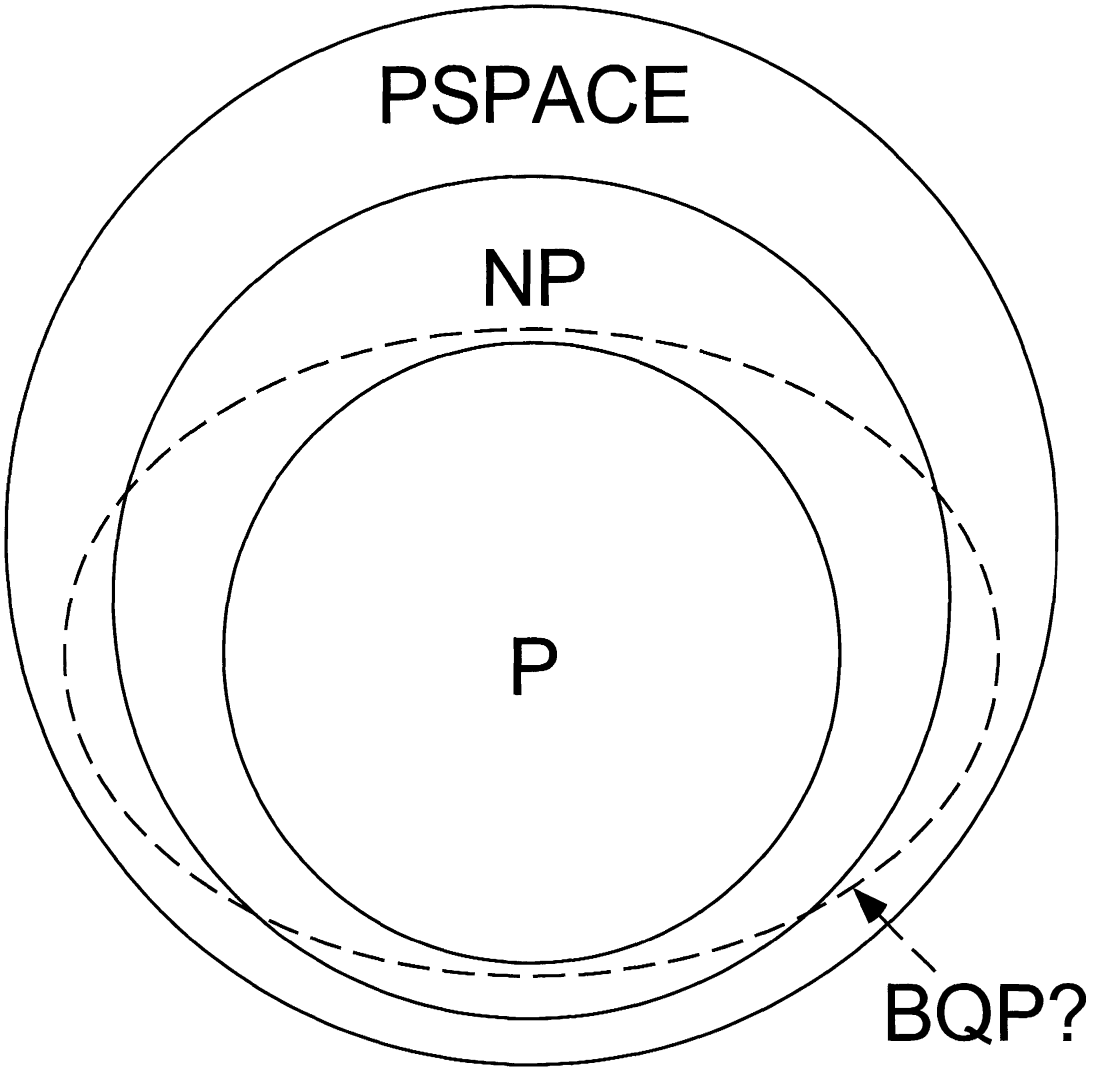


Figure 1.21. The relationship between classical and quantum complexity classes. Quantum computers can quickly solve any problem in **P**, and it is known that they can't solve problems outside of **PSPACE** quickly. Where quantum computers fit between **P** and **PSPACE** is not known, in part because we don't even know whether **PSPACE** is bigger than **P**!

We won't speculate further on the ultimate power of quantum computation now, preferring to wait until after we have better understood the principles on which fast quantum algorithms are based, a topic which occupies us for most of Part II of this book. What is already clear is that the *theory* of quantum computation poses interesting and significant challenges to the traditional notions of computation. What makes this an important challenge is that the theoretical model of quantum computation is believed to be *experimentally* realizable, because – to the best of our knowledge – this theory is consistent with the way Nature works. If this were not so then quantum computation would be just another mathematical curiosity.

1.5 Experimental quantum information processing

Quantum computation and quantum information is a wonderful theoretical discovery, but its central concepts, such as superpositions and entanglement, run counter to the intuition we garner from the everyday world around us. What evidence do we have that these ideas truly describe how Nature operates? Will the realization of large-scale quantum computers be experimentally feasible? Or might there be some principle of physics which fundamentally prohibits their eventual scaling? In the next two sections we address these questions. We begin with a review of the famous 'Stern–Gerlach' experiment, which provides evidence for the existence of qubits in Nature. We then widen our scope, addressing the broader problem of how to build practical quantum information processing systems.

1.5.1 The Stern–Gerlach experiment

The qubit is a fundamental element for quantum computation and quantum information. How do we know that systems with the properties of qubits exist in Nature? At the time of writing there is an enormous amount of evidence that this is so, but in the early days of quantum mechanics the qubit structure was not at all obvious, and people struggled with phenomena that we may now understand in terms of qubits, that is, in terms of two level quantum systems.

A decisive (and very famous) early experiment indicating the qubit structure was conceived by Stern in 1921 and performed with Gerlach in 1922 in Frankfurt. In the original Stern–Gerlach experiment, hot atoms were 'beamed' from an oven through a magnetic field which caused the atoms to be deflected, and then the position of each atom was recorded, as illustrated in Figure 1.22. The original experiment was done with silver atoms, which have a complicated structure that obscures the effects we are discussing. What we describe below actually follows a 1927 experiment done using hydrogen atoms. The same basic effect is observed, but with hydrogen atoms the discussion is easier to follow. Keep in mind, though, that this privilege wasn't available to people in the early 1920s, and they had to be very ingenious to think up explanations for the more complicated effects they observed.

Hydrogen atoms contain a proton and an orbiting electron. You can think of this electron as a little 'electric current' around the proton. This electric current causes the atom to have a magnetic field; each atom has what physicists call a 'magnetic dipole moment'. As a result each atom behaves like a little bar magnet with an axis corresponding to the axis the electron is spinning around. Throwing little bar magnets through a magnetic field causes the magnets to be deflected by the field, and we expect to see a similar deflection of atoms in the Stern–Gerlach experiment.

How the atom is deflected depends upon both the atom's magnetic dipole moment – the axis the electron is spinning around – and the magnetic field generated by the Stern–Gerlach device. We won't go through the details, but suffice to say that by constructing the Stern–Gerlach device appropriately, we can cause the atom to be deflected by an amount that depends upon the $\hat{z}$ component of the atom's magnetic dipole moment, where $\hat{z}$ is some fixed external axis.

Two major surprises emerge when this experiment is performed. First, since the hot atoms exiting the oven would naturally be expected to have their dipoles oriented randomly in every direction, it would follow that there would be a continuous distribution of atoms seen at all angles exiting from the Stern–Gerlach device. Instead, what is seen is atoms emerging from a *discrete* set of angles. Physicists were able to explain this by assuming that the magnetic dipole moment of the atoms is *quantized*, that is, comes in discrete multiples of some fundamental amount.

This observation of quantization in the Stern–Gerlach experiment was surprising to physicists of the 1920s, but not completely astonishing because evidence for quantization effects in other systems was becoming widespread at that time. What was truly surprising was the *number* of peaks seen in the experiment. The hydrogen atoms being used were such that they should have had *zero* magnetic dipole moment. Classically, this is surprising in itself, since it corresponds to no orbital motion of the electron, but based on what was known of quantum mechanics at that time this was an acceptable notion. Since the hydrogen atoms would therefore have zero magnetic moment, it was expected that only one beam of atoms would be seen, and this beam would not be deflected by the magnetic field. Instead, two beams were seen, one deflected up by the magnetic field, and the other deflected down!

This puzzling doubling was explained after considerable effort by positing that the electron in the hydrogen atom has associated with it a quantity called *spin*. This spin is not in any way associated to the usual rotational motion of the electron around the proton; it is an entirely new quantity to be associated with an electron. The great physicist Heisenberg labeled the idea 'brave' at the time it was suggested, and it is a brave idea, since it introduces an essentially new physical quantity into Nature. The spin of the electron is posited to make an *extra* contribution to the magnetic dipole moment of a hydrogen atom, in addition to the contribution due to the rotational motion of the electron.

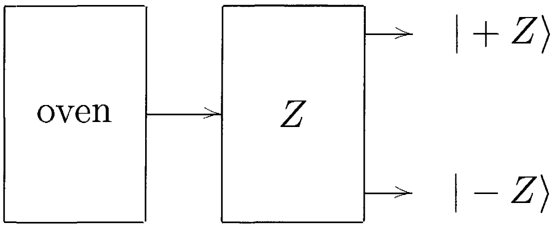


Figure 1.22. Abstract schematic of the Stern–Gerlach experiment. Hot hydrogen atoms are beamed from an oven through a magnetic field, causing a deflection either up ($|+Z\rangle$) or down ($|-Z\rangle$).

What is the proper description of the spin of the electron? As a first guess, we might hypothesize that the spin is specified by a single bit, telling the hydrogen atom to go up or down. Additional experimental results provide further useful information to determine if this guess needs refinement or replacement. Let's represent the original Stern–Gerlach apparatus as shown in Figure 1.22. Its outputs are two beams of atoms, which we shall call $|+Z\rangle$ and $|-Z\rangle$. (We're using suggestive notation which looks quantum mechanical, but of course you're free to use whatever notation you prefer.) Now suppose we cascade two Stern–Gerlach apparatus together, as shown in Figure 1.23. We arrange it so that the second apparatus is *tipped sideways*, so the magnetic field deflects atoms along the $\hat{x}$ axis. In our thought-experiment we'll block off the $|-Z\rangle$ output from the first Stern–Gerlach apparatus, while the $|+Z\rangle$ output is sent through a second apparatus oriented along the $\hat{x}$ axis. A detector is placed at the final output to measure the distribution of atoms along the $\hat{x}$ axis.

A classical magnetic dipole pointed in the $+\hat{z}$ direction has no net magnetic moment in the $\hat{x}$ direction, so we might expect that the final output would have one central peak. However, experimentally it is observed that there are two peaks of equal intensity! So perhaps these atoms are peculiar, and have definite magnetic moments along each axis, independently. That is, maybe each atom passing through the second apparatus can be

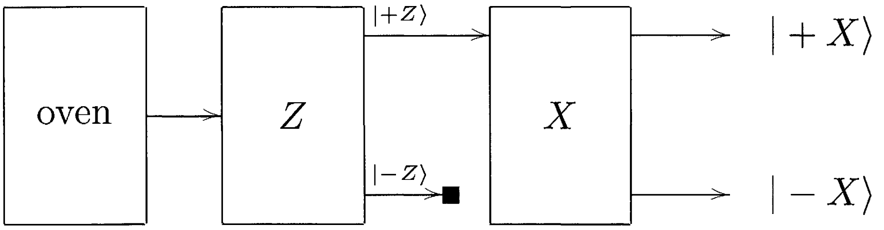


Figure 1.23. Cascaded Stern–Gerlach measurements.

described as being in a state we might write as $| + Z\rangle | + X\rangle$ or $| + Z\rangle | - X\rangle$, to indicate the two values for spin that might be observed.

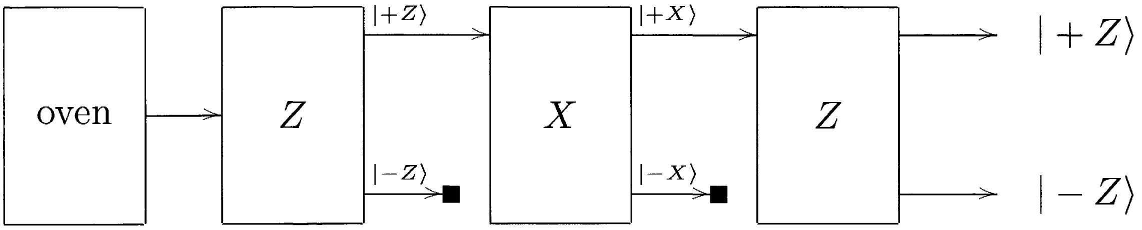


Figure 1.24. Three stage cascaded Stern–Gerlach measurements.

Another experiment, shown in Figure 1.24, can test this hypothesis by sending one beam of the previous output through a second $\hat{z}$ oriented Stern–Gerlach apparatus. If the atoms had retained their $| + Z\rangle$ orientation, then the output would be expected to have only one peak, at the $| + Z\rangle$ output. However, again *two* beams are observed at the final output, of equal intensity. Thus, the conclusion would seem to be that contrary to classical expectations, a $| + Z\rangle$ state consists of equal portions of $| + X\rangle$ and $| - X\rangle$ states, and a $| + X\rangle$ state consists of equal portions of $| + Z\rangle$ and $| - Z\rangle$ states. Similar conclusions can be reached if the Stern–Gerlach apparatus is aligned along some other axis, like the $\hat{y}$ axis.

The qubit model provides a simple explanation of this experimentally observed behavior. Let $|0\rangle$ and $|1\rangle$ be the states of a qubit, and make the assignments

$$| + Z\rangle \leftarrow |0\rangle \tag{1.56}$$

$$| - Z\rangle \leftarrow |1\rangle \tag{1.57}$$

$$| + X\rangle \leftarrow (|0\rangle + |1\rangle)/\sqrt{2}. \tag{1.58}$$

$$| - X\rangle \leftarrow (|0\rangle - |1\rangle)/\sqrt{2} \tag{1.59}$$

Then the results of the cascaded Stern–Gerlach experiment can be explained by assuming that the $\hat{z}$ Stern–Gerlach apparatus measures the spin (that is, the qubit) in the computational basis $|0\rangle, |1\rangle$, and the $\hat{x}$ Stern–Gerlach apparatus measures the spin with respect to the basis $(|0\rangle + |1\rangle)/\sqrt{2}, (|0\rangle - |1\rangle)/\sqrt{2}$. For example, in the cascaded $\hat{z}$-$\hat{x}$-$\hat{z}$ experiment, if we assume that the spins are in the state $| + Z\rangle = |0\rangle = (| + X\rangle + | - X\rangle)/\sqrt{2}$ after exiting the first Stern–Gerlach experiment, then the probability for obtaining $| + X\rangle$ out of the second apparatus is $1/2$, and the probability for $| - X\rangle$ is $1/2$. Similarly, the probability for obtaining $| + Z\rangle$ out of the third apparatus is $1/2$. A qubit model thus properly predicts results from this type of cascaded Stern–Gerlach experiment.

This example demonstrates how qubits could be a believable way of modeling systems in Nature. Of course it doesn't establish beyond all doubt that the qubit model is the correct way of understanding electron spin – far more experimental corroboration is required. Nevertheless, because of many experiments like these, we now believe that electron spin is best described by the qubit model. What is more, we believe that the qubit model (and generalizations of it to higher dimensions; quantum mechanics, in other words) is capable of describing *every* physical system. We now turn to the question of what systems are especially well adapted to quantum information processing.

1.5.2 Prospects for practical quantum information processing

Building quantum information processing devices is a great challenge for scientists and engineers of the third millennium. Will we rise to meet this challenge? Is it possible at all? Is it worth attempting? If so, how might the feat be accomplished? These are difficult and important questions, to which we essay brief answers in this section, to be expanded upon throughout the book.

The most fundamental question is whether there is any point of principle that prohibits us from doing one or more forms of quantum information processing? Two possible obstructions suggest themselves: that noise may place a fundamental barrier to useful quantum information processing; or that quantum mechanics may fail to be correct.

Noise is without a doubt a significant obstruction to the development of practical quantum information processing devices. Is it a *fundamentally irremovable* obstruction that will forever prevent the development of large-scale quantum information processing devices? The theory of quantum error-correcting codes strongly suggests that while quantum noise is a practical problem that needs to be addressed, it does not present a fundamental problem of *principle*. In particular, there is a *threshold theorem* for quantum computation, which states, roughly speaking, that provided the level of noise in a quantum computer can be reduced below a certain constant 'threshold' value, quantum error-correcting codes can be used to push it down even further, essentially *ad infinitum*, for a small overhead in the complexity of the computation. The threshold theorem makes some broad assumptions about the nature and magnitude of the noise occurring in a quantum computer, and the architecture available for performing quantum computation; however, provided those assumptions are satisfied, the effects of noise can be made essentially negligible for quantum information processing. Chapters 8, 10 and 12 discuss quantum noise, quantum error-correction and the threshold theorem in detail.

A second possibility that may preclude quantum information processing is if quantum mechanics is incorrect. Indeed, probing the validity of quantum mechanics (both relativistic and non-relativistic) is one reason for being interested in building quantum information processing devices. Never before have we explored a regime of Nature in which complete control has been obtained over large-scale quantum systems, and perhaps Nature may reveal some new surprises in this regime which are not adequately explained by quantum mechanics. If this occurs, it will be a momentous discovery in the history of science, and can be expected to have considerable consequences in other areas of science and technology, as did the discovery of quantum mechanics. Such a discovery might also impact quantum computation and quantum information; however, whether the impact would enhance, detract or not affect the power of quantum information processing cannot be predicted in advance. Until and unless such effects are found we have no way of knowing how they might affect information processing, so for the remainder of this book we go with all the evidence to date and assume that quantum mechanics is a complete and correct description of the world.

Given that there is no fundamental obstacle to building quantum information processing devices, why should we invest enormous amounts of time and money in the attempt to do so? We have already discussed several reasons for wanting to do so: practical applications such as quantum cryptography and the factoring of large composite numbers; and the desire to obtain fundamental insights into Nature and into information processing.

These are good reasons, and justify a considerable investment of time and money in the effort to build quantum information processing devices. However, it is fair to say that a clearer picture of the relative power of quantum and classical information processing is needed in order to assess their relative merits. To obtain such a picture requires further theoretical work on the foundations of quantum computation and quantum information. Of particular interest is a decisive answer to the question 'Are quantum computers more powerful than classical computers?' Even if the answer to such a question eludes us for the time being, it would be useful to have a clear path of interesting applications at varying levels of complexity to aid researchers aiming to experimentally realize quantum information processing. Historically, the advance of technology is often hastened by the use of short- to medium-term incentives as a stepping-stone to long-term goals. Consider that microprocessors were initially used as controllers for elevators and other simple devices, before graduating to be the fundamental component in personal computers (and then on to who-knows-what). Below we sketch out a path of short- to medium-term goals for people interested in achieving the long-term goal of large-scale quantum information processing.

Surprisingly many small-scale applications of quantum computation and quantum information are known. Not all are as flashy as cousins like the quantum factoring algorithm, but the relative ease of implementing small-scale applications makes them extremely important as medium-term goals in themselves.

Quantum state tomography and quantum process tomography are two elementary processes whose perfection is of great importance to quantum computation and quantum information, as well as being of independent interest in their own right. Quantum state tomography is a method for determining the quantum state of a system. To do this, it has to overcome the 'hidden' nature of the quantum state – remember, the state can't be directly determined by a measurement – by performing repeated preparations of the same quantum state, which is then measured in different ways in order to build up a complete description of the quantum state. Quantum process tomography is a more ambitious (but closely related) procedure to completely characterize the *dynamics* of a quantum system. Quantum process tomography can, for example, be used to characterize the performance of an alleged quantum gate or quantum communications channel, or to determine the types and magnitudes of different noise processes in a system. Beside obvious applications to quantum computation and quantum information, quantum process tomography can be expected to have significant applications as a diagnostic tool to aid in the evaluation and improvement of primitive operations in any field of science and technology where quantum effects are important. Quantum state tomography and quantum process tomography are described in more detail in Chapter 8.

Various small-scale communications primitives are also of great interest. We have already mentioned quantum cryptography and quantum teleportation. The former is likely to be useful in practical applications involving the distribution of a small amount of key material that needs to be highly secure. The uses of quantum teleportation are perhaps more open to question. We will see in Chapter 12 that teleportation may be an extremely useful primitive for transmitting quantum states between distant nodes in a network, in the presence of noise. The idea is to focus one's efforts on distributing EPR pairs between the nodes that wish to communicate. The EPR pairs may be corrupted during communication, but special 'entanglement distillation' protocols can then be used to 'clean up' the EPR pairs, enabling them to be used to teleport quantum states from one location to another. In fact, procotols based upon entanglement distillation and teleportation offer performance superior to more conventional quantum error-correction techniques in enabling noise free communication of qubits.

What of the medium-scale? A promising medium-scale application of quantum information processing is to the simulation of quantum systems. To simulate a quantum system containing even a few dozen 'qubits' (or the equivalent in terms of some other basic system) strains the resources of even the largest supercomputers. A simple calculation is instructive. Suppose we have a system containing 50 qubits. To describe the state of such a system requires $2^{50} \approx 10^{15}$ complex amplitudes. If the amplitudes are stored to 128 bits of precision, then it requires 256 bits or 32 bytes in order to store each amplitude, for a total of 32×10^{15} bytes of information, or about 32 thousand terabytes of information, well beyond the capacity of existing computers, and corresponding to about the storage capacity that might be expected to appear in supercomputers during the second decade of the twenty-first century, presuming that Moore's law continues on schedule. 90 qubits at the same level of precision requires 32×10^{27} bytes, which, even if implemented using single atoms to represent bits, would require kilograms (or more) of matter.

How useful will quantum simulations be? It seems likely that conventional methods will still be used to determine elementary properties of materials, such as bond strengths and basic spectroscopic properties. However, once the basic properties are well understood, it seems likely that quantum simulation will be of great utility as a laboratory for the design and testing of properties of novel molecules. In a conventional laboratory setup, many different types of 'hardware' – chemicals, detectors, and so on – may be required to test a wide variety of possible designs for a molecule. On a quantum computer, these different types of hardware can all be simulated in software, which is likely to be much less expensive and much faster. Of course, final design and testing must be performed with real physical systems; however, quantum computers may enable a much larger range of potential designs to be explored and evaluated *en route* to a better final design. It is interesting to note that such *ab initio* calculations to aid in the design of new molecules have been attempted on classical computers; however, they have met with limited success due to the enormous computational resources needed to simulate quantum mechanics on a classical computer. Quantum computers should be able to do much better in the relatively near future.

What of large-scale applications? Aside from scaling up applications like quantum simulation and quantum cryptography, relatively few large-scale applications are known: the factoring of large numbers, taking discrete logarithms, and quantum searching. Interest in the first two of these derives mainly from the *negative* effect they would have of limiting the viability of existing public key cryptographic systems. (They might also be of substantial practical interest to mathematicians interested in these problems simply for their own sake.) So it does not seem likely that factoring and discrete logarithm will be all that important as applications for the long run. Quantum searching may be of tremendous use because of the wide utility of the search heuristic, and we discuss some possible applications in Chapter 6. What would really be superb are many more large-scale applications of quantum information processing. This is a great goal for the future!

Given a path of potential applications for quantum information processing, how can it be achieved in real physical systems? At the small scale of a few qubits there are already several working proposals for quantum information processing devices. Perhaps the easiest to realize are based upon *optical* techniques, that is, electromagnetic radiation. Simple devices like mirrors and beamsplitters can be used to do elementary manipulations of photons. Interestingly, a major difficulty has been producing single photons on demand; experimentalists have instead opted to use schemes which produce single photons 'every now and then', at random, and wait for such an event to occur. Quantum cryptography, superdense coding, and quantum teleportation have all been realized using such optical techniques. A major advantage of the optical techniques is that photons tend to be highly stable carriers of quantum mechanical information. A major disadvantage is that photons don't directly interact with one another. Instead, the interaction has to be mediated by something else, like an atom, which introduces additional noise and complications into the experiment. An *effective* interaction between two photons is set up, which essentially works in two steps: photon number one interacts with the atom, which in turn interacts with the second photon, causing an overall interaction between the two photons.

An alternative scheme is based upon methods for trapping different types of atom: there is the *ion trap*, in which a small number of charged atoms are trapped in a confined space; and *neutral atom traps*, for trapping uncharged atoms in a confined space. Quantum information processing schemes based upon atom traps use the atoms to store qubits. Electromagnetic radiation also shows up in these schemes, but in a rather different way than in what we referred to as the 'optical' approach to quantum information processing. In these schemes, photons are used to manipulate the information stored in the atoms themselves, rather than as the place the information is stored. Single qubit quantum gates can be performed by applying appropriate pulses of electromagnetic radiation to individual atoms. Neighboring atoms can interact with one another via (for example) dipole forces that enable quantum gates to be accomplished. Moreover, the exact nature of the interaction between neighboring atoms can be modified by applying appropriate pulses of electromagnetic radiation to the atoms, giving the experimentalist control over what gates are performed in the system. Finally, quantum measurement can be accomplished in these systems using the long established *quantum jumps* technique, which implements with superb accuracy the measurements in the computational basis used for quantum computation.

Another class of quantum information processing schemes is based upon *Nuclear Magnetic Resonance*, often known by its initials, NMR. These schemes store quantum information in the *nuclear spin* of atoms in a molecule, and manipulate that information using electromagnetic radiation. Such schemes pose special difficulties, because in NMR it is not possible to directly access individual nuclei. Instead, a huge number (typically around 10^{15}) of essentially identical molecules are stored in solution. Electromagnetic pulses are applied to the sample, causing each molecule to respond in roughly the same way. You should think of each molecule as being an independent computer, and the sample as containing a huge number of computers all running in parallel (classically).

NMR quantum information processing faces three special difficulties that make it rather different from other quantum information processing schemes. First, the molecules are typically prepared by letting them equilibrate at room temperature, which is so much higher than typical spin flip energies that the spins become nearly completely randomly oriented. This fact makes the initial state rather more 'noisy' than is desirable for quantum information processing. How this noise may be overcome is an interesting story that we tell in Chapter 7. A second problem is that the class of measurements that may be performed in NMR falls well short of the most general measurements we would like to perform in quantum information processing. Nevertheless, for many instances of quantum information processing the class of measurements allowed in NMR is sufficient. Third, because molecules cannot be individually addressed in NMR you might ask how it is that individual qubits can be manipulated in an appropriate way. Fortunately, different nuclei in the molecule can have different properties that allow them to be individually addressed – or at least addressed at a sufficiently fine-grained scale to allow the operations essential for quantum computation.

Many of the elements required to perform large-scale quantum information processing can be found in existing proposals: superb state preparation and quantum measurements can be performed on a small number of qubits in the ion trap; superb dynamics can be performed in small molecules using NMR; fabrication technology in solid state systems allows designs to be scaled up tremendously. A single system having all these elements would be a long way down the road to a dream quantum computer. Unfortunately, all these systems are very different, and we are many, many years from having large-scale quantum computers. However, we believe that the existence of all these properties in existing (albeit different) systems does bode well for the long-term existence of large-scale quantum information processors. Furthermore, it suggests that there is a great deal of merit to pursuing *hybrid* designs which attempt to marry the best features of two or more existing technologies. For example, there is much work being done on trapping atoms inside *electromagnetic cavities*. This enables flexible manipulation of the atom inside the cavity via optical techniques, and makes possible real-time feedback control of single atoms in ways unavailable in conventional atom traps.

To conclude, note that it is important not to assess quantum information processing as though it were just another technology for information processing. For example, it is tempting to dismiss quantum computation as yet another technological fad in the evolution of the computer that will pass in time, much as other fads have passed – for example, the 'bubble memories' widely touted as the next big thing in memory during the early 1980s. This is a mistake, since quantum computation is an *abstract paradigm* for information processing that may have many *different* implementations in technology. One can compare two different proposals for quantum computing as regards their technological merits – it makes sense to compare a 'good' proposal to a 'bad' proposal – however even a very poor proposal for a quantum computer is of a different qualitative nature from a superb design for a classical computer.

1.6 Quantum information

The term 'quantum information' is used in two distinct ways in the field of quantum computation and quantum information. The first usage is as a broad catch-all for all manner of operations that might be interpreted as related to information processing using quantum mechanics. This use encompasses subjects such as quantum computation, quantum teleportation, the no-cloning theorem, and virtually all other topics in this book.

The second use of 'quantum information' is much more specialized: it refers to the study of *elementary* quantum information processing tasks. It does not typically include, for example, quantum algorithm design, since the details of specific quantum algorithms are beyond the scope of 'elementary'. To avoid confusion we will use the term 'quantum information theory' to refer to this more specialized field, in parallel with the widely used term '(classical) information theory' to describe the corresponding classical field. Of course, the term 'quantum information theory' has a drawback of its own – it might be seen as implying that theoretical considerations are all that matter! Of course, this is not the case, and experimental demonstration of the elementary processes studied by quantum information theory is of great interest.

The purpose of this section is to introduce the basic ideas of quantum information theory. Even with the restriction to elementary quantum information processing tasks, quantum information theory may look like a disordered zoo to the beginner, with many apparently unrelated subjects falling under the 'quantum information theory' rubric. In part, that's because the subject is still under development, and it's not yet clear how all the pieces fit together. However, we can identify a few fundamental goals uniting work on quantum information theory:

(1) **Identify elementary classes of static resources in quantum mechanics**. An example is the qubit. Another example is the *bit*; classical physics arises as a special case of quantum physics, so it should not be surprising that elementary static resources appearing in classical information theory should also be of great relevance in quantum information theory. Yet another example of an elementary class of static resources is a Bell state shared between two distant parties.

(2) **Identify elementary classes of dynamical processes in quantum mechanics**. A simple example is *memory*, the ability to store a quantum state over some period of time. Less trivial processes are quantum information transmission between two parties, Alice and Bob; copying (or trying to copy) a quantum state, and the process of protecting quantum information processing against the effects of noise.

(3) **Quantify resource tradeoffs incurred performing elementary dynamical processes.** For example, what are the minimal resources required to reliably transfer quantum information between two parties using a noisy communications channel?

Similar goals define classical information theory; however, quantum information theory is broader in scope than classical information theory, for quantum information theory includes all the static and dynamic elements of classical information theory, as well as *additional* static and dynamic elements.

The remainder of this section describes some examples of questions studied by quantum information theory, in each case emphasizing the fundamental static and dynamic elements under consideration, and the resource tradeoffs being considered. We begin with an example that will appear quite familiar to classical information theorists: the problem of sending classical information through a quantum channel. We then begin to branch out and explore some of the new static and dynamic processes present in quantum mechanics, such as quantum error-correction, the problem of distinguishing quantum states, and entanglement transformation. The chapter concludes with some reflections on how the tools of quantum information theory can be applied elsewhere in quantum computation and quantum information.

1.6.1 Quantum information theory: example problems

Classical information through quantum channels

The fundamental results of classical information theory are Shannon's *noiseless channel coding theorem* and Shannon's *noisy channel coding theorem.* The noiseless channel coding theorem quantifies how many bits are required to store information being emitted by a source of information, while the noisy channel coding theorem quantifies how much information can be reliably transmitted through a noisy communications channel.

What do we mean by an *information source*? Defining this notion is a fundamental problem of classical and quantum information theory, one we'll re-examine several times. For now, let's go with a provisional definition: a classical information source is described by a set of probabilities p_j, $j = 1, 2, \ldots, d$. Each use of the source results in the 'letter' j being emitted, chosen at random with probability p_j, independently for each use of the source. For instance, if the source were of English text, then the numbers j might correspond to letters of the alphabet and punctuation, with the probabilities p_j giving the relative frequencies with which the different letters appear in regular English text. Although it is not true that the letters in English appear in an independent fashion, for our purposes it will be a good enough approximation.

Regular English text includes a considerable amount of redundancy, and it is possible to exploit that redundancy to *compress* the text. For example, the letter 'e' occurs much more frequently in regular English text than does the letter 'z'. A good scheme for compressing English text will therefore represent the letter 'e' using fewer bits of information than it uses to represent 'z'. Shannon's noiseless channel coding theorem quantifies exactly how well such a compression scheme can be made to work. More precisely, the noiseless channel coding theorem tells us that a classical source described by probabilities p_j can be compressed so that on average each use of the source can be represented using $H(p_j)$ bits of information, where $H(p_j) \equiv -\sum_j p_j \log(p_j)$ is a function of the source probability distribution known as the *Shannon entropy*. Moreover, the noiseless channel coding theorem tells us that to attempt to represent the source using fewer bits than this will result in a high probability of error when the information is decompressed. (Shannon's noiseless channel coding theorem is discussed in much greater detail in Chapter 12.)

Shannon's noiseless coding theorem provides a good example where the goals of information theory listed earlier are all met. Two static resources are identified (goal number 1): the bit and the information source. A two-stage dynamic process is identified (goal 2), compressing an information source, and then decompressing to recover the information source. Finally a quantitative criterion for determining the resources consumed (goal 3) by an optimal data compression scheme is found.

Shannon's second major result, the noisy channel coding theorem, quantifies the amount of information that can be reliably transmitted through a noisy channel. In particular, suppose we wish to transfer the information being produced by some information source to another location through a noisy channel. That location may be at another point in space, or at another point in time – the latter is the problem of storing information in the presence of noise. The idea in both instances is to encode the information being produced using error-correcting codes, so that any noise introduced by the channel can be corrected at the other end of the channel. The way error-correcting codes achieve this is by introducing enough redundancy into the information sent through the channel so that even after some of the information has been corrupted it is still possible to recover the original message. For example, suppose the noisy channel is for the transmission of single bits, and the noise in the channel is such that to achieve reliable transmission each bit produced by the source must be encoded using two bits before being sent through the channel. We say that such a channel has a *capacity* of half a bit, since each use of the channel can be used to reliably convey roughly half a bit of information. Shannon's noisy channel coding theorem provides a general procedure for calculating the capacity of an arbitrary noisy channel.

Shannon's noisy channel coding theorem also achieves the three goals of information theory we stated earlier. Two types of static resources are involved (goal 1), the information source, and the bits being sent through the channel. Three dynamical processes are involved (goal 2). The primary process is the noise in the channel. To combat this noise we perform the dual processes of encoding and decoding the state in an error-correcting code. For a fixed noise model, Shannon's theorem tells us how much redundancy must be introduced by an optimal error-correction scheme if reliable information transmission is to be achieved (goal 3).

For both the noiseless and noisy channel coding theorems Shannon restricted himself to storing the output from an information source in classical systems – bits and the like. A natural question for quantum information theory is what happens if the storage medium is changed so that classical information is transmitted using quantum states as the medium. For example, it may be that Alice wishes to compress some classical information produced by an information source, transmitting the compressed information to Bob, who then decompresses it. If the medium used to store the compressed information is a quantum state, then Shannon's noiseless channel coding theorem cannot be used to determine the optimal compression and decompression scheme. One might wonder, for example, if using qubits allows a better compression rate than is possible classically. We'll study this question in Chapter 12, and prove that, in fact, qubits do not allow any significant saving in the amount of communication required to transmit information over a noiseless channel.

Naturally, the next step is to investigate the problem of transmitting classical information through a *noisy* quantum channel. Ideally, what we'd like is a result that quantifies the *capacity* of such a channel for the transmission of information. Evaluating the capacity is a very tricky job for several reasons. Quantum mechanics gives us a huge variety of noise models, since it takes place in a continuous space, and it is not at all obvious how to adapt classical error-correction techniques to combat the noise. Might it be advantageous, for example, to encode the classical information using *entangled* states, which are then transmitted one piece at a time through the noisy channel? Or perhaps it will be advantageous to decode using entangled measurements? In Chapter 12 we'll prove the *HSW (Holevo–Schumacher–Westmoreland) theorem*, which provides a lower bound on the capacity of such a channel. Indeed, it is widely believed that the HSW theorem provides an exact evaluation of the capacity, although a complete proof of this is not yet known! What remains at issue is whether or not encoding using entangled states can be used to raise the capacity beyond the lower bound provided by the HSW theorem. All evidence to date suggests that this doesn't help raise the capacity, but it is still a fascinating open problem of quantum information theory to determine the truth or falsity of this conjecture.

Quantum information through quantum channels

Classical information is, of course, not the only static resource available in quantum mechanics. Quantum states themselves are a natural static resource, even more natural than classical information. Let's look at a *different* quantum analogue of Shannon's coding theorems, this time involving the compression and decompression of quantum states.

To begin, we need to define some quantum notion of an information source, analogous to the classical definition of an information source. As in the classical case, there are several different ways of doing this, but for the sake of definiteness let's make the provisional definition that a quantum source is described by a set of probabilities p_j and corresponding quantum states $|\psi_j\rangle$. Each use of the source produces a state $|\psi_j\rangle$ with probability p_j, with different uses of the source being independent of one another.

Is it possible to compress the output from such a quantum mechanical source? Consider the case of a qubit source which outputs the state $|0\rangle$ with probability p and the state $|1\rangle$ with probability $1 - p$. This is essentially the same as a classical source emitting single bits, either 0 with probability p, or 1 with probability $1 - p$, so it is not surprising that similar techniques can be used to compress the source so that only $H(p, 1 - p)$ qubits are required to store the compressed source, where $H(\cdot)$ is again the Shannon entropy function.

What if the source had instead been producing the state $|0\rangle$ with probability p, and the state $(|0\rangle + |1\rangle)/\sqrt{2}$ with probability $1 - p$? The standard techniques of classical data compression no longer apply, since in general it is not possible for us to distinguish the states $|0\rangle$ and $(|0\rangle + |1\rangle)/\sqrt{2}$. Might it still be possible to perform some type of compression operation?

It turns out that a type of compression is still possible, even in this instance. What is interesting is that the compression may no longer be *error-free*, in the sense that the quantum states being produced by the source may be slightly distorted by the compression–decompression procedure. Nevertheless, we require that this distortion ought to become very small and ultimately negligible in the limit of large blocks of source output being compressed. To quantify the distortion we introduce a *fidelity* measure for the compression scheme, which measures the average distortion introduced by the compression scheme. The idea of quantum data compression is that the compressed data should be recovered with very good fidelity. Think of the fidelity as being analogous to the probability of doing the decompression correctly – in the limit of large block lengths, it should tend towards the no error limit of 1.

Schumacher's noiseless channel coding theorem quantifies the resources required to do quantum data compression, with the restriction that it be possible to recover the source with fidelity close to 1. In the case of a source producing orthogonal quantum states $|\psi_j\rangle$ with probabilities p_j Schumacher's theorem reduces to telling us that the source may be compressed down to but not beyond the classical limit $H(p_j)$. However, in the more general case of non-orthogonal states being produced by the source, Schumacher's theorem tells us how much a quantum source may be compressed, and the answer is *not* the Shannon entropy $H(p_j)$! Instead, a new entropic quantity, the *von Neumann* entropy, turns out to be the correct answer. In general, the von Neumann entropy agrees with the Shannon entropy if and only if the states $|\psi_j\rangle$ are orthogonal. Otherwise, the von Neumann entropy for the source $p_j, |\psi_j\rangle$ is in general strictly *smaller* than the Shannon entropy $H(p_j)$. Thus, for example, a source producing the state $|0\rangle$ with probability p

and $(|0\rangle + |1\rangle)/\sqrt{2}$ with probability $1 - p$ can be reliably compressed using fewer than $H(p, 1 - p)$ qubits per use of the source!

The basic intuition for this decrease in resources required can be understood quite easily. Suppose the source emitting states $|0\rangle$ with probability p and $(|0\rangle + |1\rangle)/\sqrt{2}$ with probability $1 - p$ is used a large number n times. Then by the law of large numbers, with high probability the source emits about np copies of $|0\rangle$ and $n(1 - p)$ copies of $(|0\rangle + |1\rangle)/\sqrt{2}$. That is, it has the form

$$|0\rangle^{\otimes np}\left(\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right)^{\otimes n(1-p)}, \tag{1.60}$$

up to re-ordering of the systems involved. Suppose we expand the product of $|0\rangle + |1\rangle$ terms on the right hand side. Since $n(1 - p)$ is large, we can again use the law of large numbers to deduce that the terms in the product will be roughly one-half $|0\rangle$s and one-half $|1\rangle$s. That is, the $|0\rangle + |1\rangle$ product can be well approximated by a superposition of states of the form

$$|0\rangle^{\otimes n(1-p)/2}|1\rangle^{\otimes n(1-p)/2}. \tag{1.61}$$

Thus the state emitted by the source can be approximated as a superposition of terms of the form

$$|0\rangle^{\otimes n(1+p)/2}|1\rangle^{\otimes n(1-p)/2}. \tag{1.62}$$

How many states of this form are there? Roughly n choose $n(1 + p)/2$, which by Stirling's approximation is equal to $N \equiv 2^{nH[(1+p)/2,(1-p)/2]}$. A simple compression method then is to label all states of the form (1.62) $|c_1\rangle$ through $|c_N\rangle$. It is possible to perform a unitary transform on the n qubits emitted from the source that takes $|c_j\rangle$ to $|j\rangle|0\rangle^{\otimes n-nH[(1+p)/2,(1-p)/2]}$, since j is an $nH[(1 + p)/2, (1 - p)/2]$ bit number. The compression operation is to perform this unitary transformation, and then drop the final $n - nH[(1 + p)/2, (1 - p)/2]$ qubits, leaving a compressed state of $nH[(1 + p)/2, (1 - p)/2]$ qubits. To decompress we append the state $|0\rangle^{\otimes n-nH[(1+p)/2,(1-p)/2]}$ to the compressed state, and perform the inverse unitary transformation.

This procedure for quantum data compression and decompression results in a storage requirement of $H[(1 + p)/2, (1 - p)/2]$ qubits per use of the source, which whenever $p \geq 1/3$ is an improvement over the $H(p, 1 - p)$ qubits we might naively have expected from Shannon's noiseless channel coding theorem. In fact, Schumacher's noiseless channel coding theorem allows us to do somewhat better even than this, as we will see in Chapter 12; however, the essential reason in that construction is the same as the reason we were able to compress here: we exploited the fact that $|0\rangle$ and $(|0\rangle + |1\rangle)/\sqrt{2}$ are not orthogonal. Intuitively, the states contain some redundancy since both have a component in the $|0\rangle$ direction, which results in more physical similarity than would be obtained from orthogonal states. It is this redundancy that we have exploited in the coding scheme just described, and which is used in the full proof of Schumacher's noiseless channel coding theorem. Note that the restriction $p \geq 1/3$ arises because when $p < 1/3$ this particular scheme doesn't exploit the redundancy in the states: we end up effectively *increasing* the redundancy present in the problem! Of course, this is an artifact of the particular scheme we have chosen, and the general solution exploits the redundancy in a much more sensible way to achieve data compression.

Schumacher's noiseless channel coding theorem is an analogue of Shannon's noiseless channel coding theorem for the compression and decompression of quantum states. Can we find an analogue of Shannon's noisy channel coding theorem? Considerable progress on this important question has been made, using the theory of quantum error-correcting codes; however, a fully satisfactory analogue has not yet been found. We review some of what is known about the quantum channel capacity in Chapter 12.

Quantum distinguishability

Thus far all the dynamical processes we have considered – compression, decompression, noise, encoding and decoding error-correcting codes – arise in both classical and quantum information theory. However, the introduction of new types of information, such as quantum states, enlarges the class of dynamical processes beyond those considered in classical information theory. A good example is the problem of distinguishing quantum states. Classically, we are used to being able to distinguish different items of information, at least in principle. In practice, of course, a smudged letter 'a' written on a page may be very difficult to distinguish from a letter 'o', but in principle it is possible to distinguish between the two possibilities with perfect certainty.

On the other hand, quantum mechanically it is *not* always possible to distinguish between arbitrary states. For example, there is no process allowed by quantum mechanics that will reliably distinguish between the states $|0\rangle$ and $(|0\rangle + |1\rangle)/\sqrt{2}$. Proving this rigorously requires tools we don't presently have available (it is done in Chapter 2), but by considering examples it's pretty easy to convince oneself that it is not possible. Suppose, for example, that we try to distinguish the two states by measuring in the computational basis. Then, if we have been given the state $|0\rangle$, the measurement will yield 0 with probability 1. However, when we measure $(|0\rangle + |1\rangle)/\sqrt{2}$ the measurement yields 0 with probability 1/2 and 1 with probability 1/2. Thus, while a measurement result of 1 implies that state must have been $(|0\rangle + |1\rangle)/\sqrt{2}$, since it couldn't have been $|0\rangle$, we can't infer anything about the identity of the quantum state from a measurement result of 0.

This indistinguishability of non-orthogonal quantum states is at the heart of quantum computation and quantum information. It is the essence of our assertion that a quantum state contains hidden information that is not accessible to measurement, and thus plays a key role in quantum algorithms and quantum cryptography. One of the central problems of quantum information theory is to develop measures quantifying how well non-orthogonal quantum states may be distinguished, and much of Chapters 9 and 12 is concerned with this goal. In this introduction we'll limit ourselves to pointing out two interesting aspects of indistinguishability – a connection with the possibility of faster-than-light communication, and an application to 'quantum money.'

Imagine for a moment that we could distinguish between arbitrary quantum states. We'll show that this implies the ability to communicate faster than light, using entanglement. Suppose Alice and Bob share an entangled pair of qubits in the state $(|00\rangle + |11\rangle)/\sqrt{2}$. Then, if Alice measures in the computational basis, the post-measurement states will be $|00\rangle$ with probability 1/2, and $|11\rangle$ with probability 1/2. Thus Bob's system is either in the state $|0\rangle$, with probability 1/2, or in the state $|1\rangle$, with probability 1/2. Suppose, however, that Alice had instead measured in the $|+\rangle, |-\rangle$ basis. Recall that $|0\rangle = (|+\rangle + |-\rangle)/\sqrt{2}$ and $|1\rangle = (|+\rangle - |-\rangle)/\sqrt{2}$. A little algebra shows that the initial state of Alice and Bob's system may be rewritten as $(| + +\rangle + | - -\rangle)/\sqrt{2}$. Therefore, *if Alice measures in the* $|+\rangle, |-\rangle$ *basis, the state of Bob's system after the measurement* will be $|+\rangle$ or $|-\rangle$ with probability 1/2 each. So far, this is all basic quantum mechanics. But if Bob had access to a device that could distinguish the four states $|0\rangle, |1\rangle, |+\rangle, |-\rangle$ from one another, then he could tell whether Alice had measured in the computational basis, or in the $|+\rangle, |-\rangle$ basis. Moreover, he could get that information *instantaneously*, as soon as Alice had made the measurement, providing a means by which Alice and Bob could achieve faster-than-light communication! Of course, we know that it is not possible to distinguish non-orthogonal quantum states; this example shows that this restriction is also intimately tied to other physical properties which we expect the world to obey.

The indistinguishability of non-orthogonal quantum states need not always be a handicap. Sometimes it can be a boon. Imagine that a bank produces banknotes imprinted with a (classical) serial number, and a sequence of qubits each in either the state $|0\rangle$ or $(|0\rangle + |1\rangle)/\sqrt{2}$. Nobody but the bank knows what sequence of these two states is embedded in the note, and the bank maintains a list matching serial numbers to embedded states. The note is impossible to counterfeit exactly, because it is impossible for a would-be counterfeiter to determine with certainty the state of the qubits in the original note, without destroying them. When presented with the banknote a merchant (of certifiable repute) can verify that it is not a counterfeit by calling the bank, telling them the serial number, and then asking what sequence of states were embedded in the note. They can then check that the note is genuine by measuring the qubits in the $|0\rangle, |1\rangle$ or $(|0\rangle + |1\rangle)/\sqrt{2}, (|0\rangle - |1\rangle)/\sqrt{2}$ basis, as directed by the bank. With probability which increases exponentially to one with the number of qubits checked, any would-be counterfeiter will be detected at this stage! This idea is the basis for numerous other quantum cryptographic protocols, and demonstrates the utility of the indistinguishability of non-orthogonal quantum states.

Exercise 1.2: Explain how a device which, upon input of one of two non-orthogonal quantum states $|\psi\rangle$ or $|\varphi\rangle$ correctly identified the state, could be used to build a device which cloned the states $|\psi\rangle$ and $|\varphi\rangle$, in violation of the no-cloning theorem. Conversely, explain how a device for cloning could be used to distinguish non-orthogonal quantum states.

Creation and transformation of entanglement

Entanglement is another elementary static resource of quantum mechanics. Its properties are amazingly different from those of the resources most familiar from classical information theory, and they are not yet well understood; we have at best an incomplete collage of results related to entanglement. We don't yet have all the language needed to understand the solutions, but let's at least look at two information-theoretic problems related to entanglement.

Creating entanglement is a simple dynamical process of interest in quantum information theory. How many qubits must two parties exchange if they are to create a particular entangled state shared between them, given that they share no prior entanglement? A second dynamical process of interest is *transforming entanglement* from one form into another. Suppose, for example, that Alice and Bob share between them a Bell state, and wish to transform it into some other type of entangled state. What resources do they need to accomplish this task? Can they do it without communicating? With classical communication only? If quantum communication is required then how much quantum communication is required?

Answering these and more complex questions about the creation and transformation of entanglement forms a fascinating area of study in its own right, and also promises to give insight into tasks such as quantum computation. For example, a distributed quantum computation may be viewed as simply a method for generating entanglement between two or more parties; lower bounds on the amount of communication that must be done to perform such a distributed quantum computation then follow from lower bounds on the amount of communication that must be performed to create appropriate entangled states.

1.6.2 Quantum information in a wider context

We have given but the barest glimpse of quantum information theory. Part III of this book discusses quantum information theory in much greater detail, especially Chapter 11, which deals with fundamental properties of entropy in quantum and classical information theory, and Chapter 12, which focuses on pure quantum information theory.

Quantum information theory is the most abstract part of quantum computation and quantum information, yet in some sense it is also the most fundamental. The question driving quantum information theory, and ultimately all of quantum computation and quantum information, is *what makes quantum information processing tick?* What is it that separates the quantum and the classical world? What resources, unavailable in a classical world, are being utilized in a quantum computation? Existing answers to these questions are foggy and incomplete; it is our hope that the fog may yet lift in the years to come, and we will obtain a clear appreciation for the possibilities and limitations of quantum information processing.

Problem 1.1: (Feynman-Gates conversation) Construct a friendly imaginary discussion of about 2000 words between Bill Gates and Richard Feynman, set in the present, on the future of computation. (*Comment*: You might like to try waiting until you've read the rest of the book before attempting this question. See the 'History and further reading' below for pointers to one possible answer for this question.)

Problem 1.2: What is the most significant discovery yet made in quantum computation and quantum information? Write an essay of about 2000 words for an educated lay audience about the discovery. (*Comment*: As for the previous problem, you might like to try waiting until you've read the rest of the book before attempting this question.)

History and further reading

Most of the material in this chapter is revisited in more depth in later chapters. Therefore the historical references and further reading below are limited to material which does not recur in later chapters.

Piecing together the historical context in which quantum computation and quantum information have developed requires a broad overview of the history of many fields. We have tried to tie this history together in this chapter, but inevitably much background material was omitted due to limited space and expertise. The following recommendations attempt to redress this omission.

The history of quantum mechanics has been told in many places. We recommend especially the outstanding works of Pais[Pai82, Pai86, Pai91]. Of these three, [Pai86] is most directly concerned with the development of quantum mechanics; however, Pais' biographies of Einstein[Pai82] and of Bohr[Pai91] also contain much material of interest, at a less intense level. The rise of technologies based upon quantum mechanics has been described by Milburn[Mil97, Mil98]. Turing's marvelous paper on the foundations of computer science[Tur36] is well worth reading. It can be found in the valuable historical collection of Davis[Dav65]. Hofstadter[Hof79] and Penrose[Pen89] contain entertaining and informative discussions of the foundations of computer science. Shasha and Lazere's biography of fifteen leading computer scientists[SL98] gives considerable insight into many different facets of the history of computer science. Finally, Knuth's awesome series of books[Knu97, Knu98a, Knu98b] contain an amazing amount of historical information. Shannon's brilliant papers founding information theory make excellent reading[Sha48] (also reprinted in [SW49]). MacWilliams and Sloane[MS77] is not only an excellent text on error-correcting codes, but also contains an enormous amount of useful historical information. Similarly, Cover and Thomas[CT91] is an excellent text on information theory, with extensive historical information. Shannon's collected works, together with many useful historical items have been collected in a large volume[SW93] edited by Sloane and Wyner. Slepian has also collected a useful set of reprints on information theory[Sle74]. Cryptography is an ancient art with an intricate and often interesting history. Kahn[Kah96] is a huge history of cryptography containing a wealth of information. For more recent developments we recommend the books by Menezes, van Oorschot, and Vanstone[MvOV96], Schneier[Sch96a], and by Diffie and Landau[DL98].

Quantum teleportation was discovered by Bennett, Brassard, Crépeau, Jozsa, Peres, and Wootters[BBC+93], and later experimentally realized in various different forms by Boschi, Branca, De Martini, Hardy and Popescu[BBM+98] using optical techniques, by Bouwmeester, Pan, Mattle, Eibl, Weinfurter, and Zeilinger[BPM+97] using photon polarization, by Furusawa, Sørensen, Braunstein, Fuchs, Kimble, and Polzik using 'squeezed' states of light[FSB+98], and by Nielsen, Knill, and Laflamme using NMR[NKL98].

Deutsch's problem was posed by Deutsch[Deu85], and a one-bit solution was given in the same paper. The extension to the general n-bit case was given by Deutsch and Jozsa[DJ92]. The algorithms in these early papers have been substantially improved subsequently by Cleve, Ekert, Macchiavello, and Mosca[CEMM98], and independently in unpublished work by Tapp. In this chapter we have given the improved version of the algorithm, which fits very nicely into the hidden subgroup problem framework that will later be discussed in Chapter 5. The original algorithm of Deutsch only worked probabilistically; Deutsch and Jozsa improved this to obtain a deterministic algorithm, but their method required two function evaluations, in contrast to the improved algorithms presented in this chapter. Nevertheless, it is still conventional to refer to these algorithms as Deutsch's algorithm and the Deutsch–Jozsa algorithm in honor of two huge leaps forward: the concrete demonstration by Deutsch that a quantum computer could do something faster than a classical computer; and the extension by Deutsch and Jozsa which demonstrated for the first time a similar gap for the scaling of the time required to solve a problem.

Excellent discussions of the Stern–Gerlach experiment can be found in standard quantum mechanics textbooks such as the texts by Sakurai[Sak95], Volume III of Feynman, Leighton and Sands[FLS65a], and Cohen-Tannoudji, Diu and Laloë[CTDL77a, CTDL77b].

Problem 1.1 was suggested by the lovely article of Rahim[Rah99].