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 to 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 an 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 counter-intuitive, 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 quantum bits (or *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. The Solovay-Strassen test was of especial significance at the time it was proposed as no deterministic test for primality was then known, nor is one known at the time of this writing. 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, discov-

ered 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

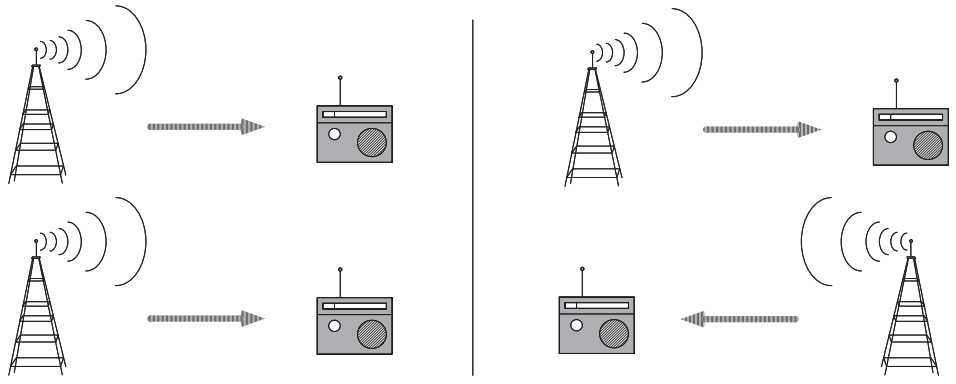

Figure 1.1. Classically, if we have two very noisy channels of zero capacity running side by side, then the combined channel has zero capacity to send information. Not surprisingly, if we reverse the direction of one of the channels, we still have zero capacity to send information. Quantum mechanically,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