When Abandoned Bits Bite Back – Reversibility and Quantum Computing

Paul Fitzpatrick

AI Lab, MIT, Cambridge, USA
paulfitz@ai.mit.edu

Abstract. Reversible logic was originally proposed as a way to build classical computers with very low energy requirements. Its greatest impact so far has in fact been on the field of quantum computing, where it is used pervasively – but for an entirely different purpose. Quantum interference effects offer the potential for speeding up certain classical calculations. But if state information is expelled from a quantum computer into the unmodeled environment – as happens whenever an irreversible operation is performed – then it becomes impossible to predict or control when interference will occur. Reversibility is necessary because it allows bits to be erased by uncomputing them rather than simply expelling them, preserving control over interference effects.

Introduction

Whenever a bit in a computer is reset to a fixed value, the entropy of the machine's logical state decreases. No matter how efficient the computer is, the second law of thermodynamics guarantees that this local reduction in entropy must come at a price – normally paid as a small amount of heat dissipation elsewhere (Landauer 1961). Reversible logic offers a way to compute without changing the entropy of the computer's state, and so potentially avoiding the energy wasted on such heat dissipation. This is important for the continued development of classical computers, since continued miniaturization is fast approaching the scale which this form of energy loss could be significant. And even now lessons drawn from reversible logic are having an impact (Gershenfeld 1996).

Quantum computing is a developing technology that seeks to harness quantum interference effects for computational purposes (Deutsch 1985). It completely foregoes use of the irreversible gates upon which classical computers are based, such as NAND, and instead exclusively uses reversible operators. If the only argument in favor of reversible logic was minimizing energy consumption, then this policy would be very ill-advised. Quantum computing is an immature technology (Preskill 1998a), facing so many challenges that the energy consumption associated with erasing a bit is a complete non-issue – at least until such time as it no longer requires an entire physics laboratory to run a computation. In fact there is a much stronger reason for avoiding irreversible operations. Quantum systems can evolve in superpositions of classical states, and by making these states interfere constructively and destructively useful computation can be performed (Chuang, Laflamme, Shor & Zurek 1995). If we try to erase a quantum bit (or qubit) that is involved in a superposition, we end up dispersing its state into the uncontrolled degrees of freedom of the computer. This is likely to be disastrous, because the superposition will "follow" the state out into the environment and no longer be under the programmer's full control. So operations on the state of a quantum computer must be reversible while the conditions for interference are being put in place, otherwise the results will at best be no more than what a classical computer could achieve using the same number of operations, and at worst complete garbage.

The goal of this paper is to trace the consequences reversibility has for quantum computing. Section 1 reviews the case for determinism and reversibility in classical and quantum physics. Section 2 introduces the mathematical representation of quantum states, and shows how erasing a quantum bit by discarding it into the environment is asking for trouble. Section 3 introduces classical reversible logic, as first developed by (Bennett 1973), establishing that useful computation can be done within the constraints of reversibility. For a

quantum system, the class of reversible operations is in fact far richer than it is classically. Section 4 explores the evolution of a quantum system, and Section 5 shows how this this evolution can be engineered in terms of the quantum analogues of classical gates, following the constructions introduced by Barenco et al. (1995).

All of this theory relies on completely shielding the state of the quantum computer from its environment. But it is easy to show that a single stray interaction is enough to completely disrupt a computation. Until a few years ago, it was not clear whether quantum states could be protected even in principle. But eventually novel methods were found to implement error correction (Section 6) and fault tolerant computation (Section 7). In effect these procedures create a "refrigerator" to keep the computer at constant entropy and pump away disturbances. The requirements of fault tolerance greatly affect how quantum gates should be constructed. This paper ends by examining the gate design methodology proposed by Gottesman & Chuang (1999), which borrows from the method used to implement quantum teleportation (Section 8) to build gates that are compatible with the principles of fault tolerance. Their design, presented in simplified form in Section 9, is a good example of how irreversible processes can be used to support a reversible computation.

1 Determinism and Reversibility

A deterministic process is one whose future evolution is completely determined by its current state. A deterministic process is reversible if the current state also uniquely determines its own past history (see Figure 1). The laws of physics are, as far as we know, completely reversible at the microscopic level. Most physical processes are equally consistent with known law when the flow of time is inverted, and so their current state clearly uniquely determines both their past and future. A small number of processes are suspected to violate this simple time-reversal symmetry, such as the decay of neutral kaons (Angelopoulos et al. 1998). But a stronger symmetry called CPT invariance (where time reversal is augmented with charge conjugation and parity inversion) implies that these processes are reversible as well. Extremes of physics such as black holes might be able to dent reversibility, but this is far from established, and is not yet relevant to a discussion of computation. Of course, at a macroscopic scale physics often appears irreversible. Once particles are being described at a statistical level and their detailed individual trajectories are no longer tracked, then irreversible models are indeed appropriate and necessary.

Reversibility has important implications for the physics of computation. It is commonplace to apply irreversible operations to the logical state space of a computer, by resetting bits to zero or applying gates with fewer outputs than inputs (AND, OR, NAND, and so on). Reversibility of physics tells us that this logical state space cannot map directly onto a closed physical implementation. Implementing such irreversible operations must necessarily involve exporting some state information into the unmodeled environment of the computer, preserving global physical reversibility while permitting logical irreversibility of the gate.

But are quantum processes reversible? To be reversible, they must first be deterministic. Quantum systems are to the contrary often asserted to be nondeterministic and probabilistic in nature. But this is in truth more a statement of how much we can measure about the state of a system than how well we could predict its evolution

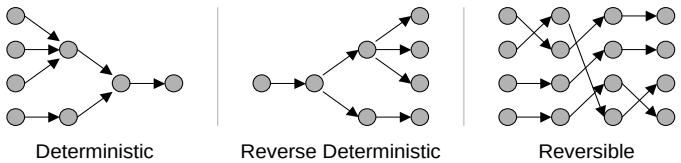


Fig. 1. States may have a unique future (left) or a unique past (center). If they have both, the evolution of the system is reversible (right).

if its state were completely known. Quantum processes as currently understood can easily be interpreted as fully deterministic and reversible when full state information is available, as the next section will illustrate. Claims of nondeterminism in quantum mechanics largely stem from confusion about the status of measurement in that theory.

It is true that there are unusual constraints on how much information about the state of a quantum system is accessible to measurement. This might suggest that even if quantum theory is deterministic in principle, it is in practice nondeterministic because we cannot acquire all the information we need to predict a system's evolution. This view has some merit – but in the context of computation, we are responsible both for preparing the initial state of the computer and choosing what degrees of freedom that state is stored in, and so we can easily arrange to know everything needed to fully predict the evolution of the computer. The next section tries to clarify the nature of quantum states, both fully and partially known.

2 Quantum states

To be able to predict the future evolution of a physical system, we need to know its current state. It is a postulate of quantum mechanics that the state of a closed quantum system can be described as a unit vector in a Hilbert space (i.e. a vector space with an inner product). The states of the system that are fully distinguishable from each other by measurement form a basis for that space, rather than comprising the space in its entirety as they do classically. For example, the state space associated with the spin of an election is a two-dimensional complex vector space, with the distinct "spin-down" and "spin-up" states of the electron forming an orthonormal basis for the space. If we choose to write these states as $|0\rangle$ and $|1\rangle$, where the traditional "ket" notation $|x\rangle$ just denotes a column vector named "x", then vectors in this space may be written as:

$$|\psi\rangle = a|0\rangle + b|1\rangle$$

where a and b are complex numbers. A system like this, with two fully distinguishable states, is called a qubit. The inner product of two such vectors is written as $\langle\psi|\phi\rangle$ where:

$$\begin{aligned}\langle\psi|\phi\rangle &= |\psi\rangle^\dagger|\phi\rangle \\ &= (|\psi\rangle^*)^T|\phi\rangle\end{aligned}$$

which is just the usual inner product for a complex vector space. The vector $|\psi\rangle^\dagger$ is written as $\langle\psi|$, using the "bra" notation, and is the conjugate transpose of $|\psi\rangle$, and therefore a row vector. Since this notation will be used throughout the paper, here is an example of it in use:

$$\begin{aligned}\langle\psi|\psi\rangle &= (a|0\rangle + b|1\rangle)^\dagger\ (a|0\rangle + b|1\rangle) \\ &= (a^*\langle 0| + b^*\langle 1|)\ (a|0\rangle + b|1\rangle) \\ &= a^*a\langle 0|0\rangle + a^*b\langle 0|1\rangle + b^*a\langle 1|0\rangle + b^*b\langle 1|1\rangle \\ &= a^*a + b^*b = |a|^2 + |b|^2\end{aligned}$$

This shows that the requirement that the state vector be of unit length implies that $|a|^2 + |b|^2 = 1$. We will later see that these squared amplitudes correspond to the probabilities of the spin being found in one of the basis states after an appropriate measurement – so it is natural that they should sum to one. An example of a non-classical state is:

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

where the notation $|+\rangle$ is simply a common name for this state.

Composition of states

The state space associated with a collection of quantum systems is the tensor product of the state spaces of its components. Suppose we have a system composed of two spin particles, A and B. Let $|0\rangle_A$ and $|1\rangle_A$ be a basis for the state space of particle A, and $|0\rangle_B$ and $|1\rangle_B$ be a basis for the state space of particle B. Then the composite system is modeled as a four-dimensional vector space with four basis vectors:

$$\{\ |0\rangle_A \otimes |0\rangle_B,\ \ |0\rangle_A \otimes |1\rangle_B,\ \ |1\rangle_A \otimes |0\rangle_B,\ \ |1\rangle_A \otimes |1\rangle_B\ \}$$

These basis vectors have a simple interpretation in terms of the individual particles that make up the system. For example, the vector $|1\rangle_A \otimes |0\rangle_B$ describes a state where particle A is in state $|1\rangle_A$ and particle B is in state $|0\rangle_B$. These basis vectors correspond to the four fully distinguishable states of the two-particle system, and are exactly the ones we would have picked out if we tried to represent the composite system directly, rather than building up from its components. Basis vectors of this nature are very important in quantum computing, so their notation is often shortened. The state $|1\rangle_A \otimes |0\rangle_B$ may also be written as $|10\rangle_{AB}$, or $|10\rangle$, or even just $|2\rangle$ when doing so does not introduce ambiguity. The state of the combined system can be written in terms of the basis vectors as:

$$|\psi\rangle = a|00\rangle + b|01\rangle + c|10\rangle + d|11\rangle$$

where a, b, c, and d are complex numbers, with $|a|^2 + |b|^2 + |c|^2 + |d|^2 = 1$. The state space of this composite system has some counter-intuitive features. Some composite states are easy to interpret in terms of states of the individual particles, but many are not. An example of an easy state to interpret is:

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

So this state of the composite system corresponds simply to A being in the state $|+\rangle$ and B being in the state $|-\rangle$. A state that can be factored in this way is called a "product" state. But now consider the state:

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

This is a perfectly legal state, called a Bell state. It is a unit vector in the Hilbert space associated with the two spin particles. Yet there are no states of the individual spins which together give this state – it is not a product state. Such states are called "entangled". We will see later how these states can be produced, and how to put them to work. For example, the state $|\beta_{00}\rangle$ will turn out to be key to implementing quantum teleportation.

Mixtures of states

The states we have described so far are called "pure" states – they are known completely. This section takes a first step towards modeling states that are only partially known. Suppose we randomly prepare a spin particle to be in either the $|0\rangle$ or $|1\rangle$ state, with equal probability, using a random number generator. The particle is then said to be in a "mixed" state. If we prepare a large number of such particles in this way, then it turns out that any measurements we make on them will be statistically indistinguishable from another collection prepared in either the $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ or $\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$ state using the same procedure. And both of these situations are also statistically indistinguishable from another collection with each particle prepared in any of the four states

$a|0\rangle \pm b|1\rangle$ or $b|0\rangle \pm a|1\rangle$ with equal probability, for any a and b. The *density operator* representation is a tool that allows mixtures to be expressed in a form that makes it obvious which are distinguishable and which are not. Suppose the probability of the system being in one of a set of states $|\psi_i\rangle$ is p_i. Then we can write the state of the system in terms of a density operator as follows:

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

where $|\psi\rangle\langle\phi|$ represents the outer product of the vectors $|\psi\rangle$ and $|\phi\rangle$. The density operator is useful because different distributions of states can give rise to the same measurement statistics, and if they do they will have the same density matrix. So the density matrix hides distinctions that have no measurable consequences. For example, an equal-probability mixture of $|0\rangle$ and $|1\rangle$ gives:

$$\rho = \frac{1}{2}|0\rangle\langle 0| + \frac{1}{2}|1\rangle\langle 1|$$

and an equal-probability mixture of $|+\rangle$ and $|-\rangle$ gives:

$$\begin{aligned}
\rho' &= \frac{1}{2}\left(\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)\frac{1}{\sqrt{2}}(\langle 0| + \langle 1|)\right) + \frac{1}{2}\left(\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)\frac{1}{\sqrt{2}}(\langle 0| - \langle 1|)\right) \\
&= \frac{1}{4}\left(|0\rangle\langle 0| + |0\rangle\langle 1| + |1\rangle\langle 0| + |1\rangle\langle 1|\right) + \frac{1}{4}\left(|0\rangle\langle 0| - |0\rangle\langle 1| - |1\rangle\langle 0| + |1\rangle\langle 1|\right) \\
&= \frac{1}{2}|0\rangle\langle 0| + \frac{1}{2}|1\rangle\langle 1|
\end{aligned}$$

which is the same as for the mixture of $|0\rangle$ and $|1\rangle$.

Decomposition of states

Suppose we take a system whose state is known and then physically divide it into isolated subsystems. Classically, we would them be able to predict the evolution of each subsystem individually. For a quantum system, the situation is more complex. We saw earlier an example of a pure state of a two-particle system that cannot be expressed as independent states of the individual particles. The evolution of one of the particles cannot be completely predicted once information about the other is removed. So a local model will not be deterministic. In fact it turns out that for a quantum system the local model has a probabilistic interpretation based on mixed states, conveniently expressed using the density operator introduced in the previous section. Given two systems A and B whose state together is described by the density operator ρ^{AB}, we can compute the *reduced density operator* for system A alone as:

$$\rho^A = \mathrm{tr}_B(\rho^{AB})$$

where tr_B is the partial trace over system B:

$$\mathrm{tr}_B(|a_0\rangle\langle a_1| \otimes |b_0\rangle\langle b_1|) = |a_0\rangle\langle a_1|\mathrm{tr}(|b_0\rangle\langle b_1|)$$

Let us look at what we can say about an individual particle in the entangled two-particle state $|\beta_{00}\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ considered earlier. The density operator for this system will be:

$$\begin{aligned}
\rho &= |\beta_{00}\rangle\langle\beta_{00}| \\
&= \frac{1}{2}(|00\rangle + |11\rangle)(\langle 00| + \langle 11|) \\
&= \frac{1}{2}\left(|00\rangle\langle 00| + |00\rangle\langle 11| + |11\rangle\langle 00| + |11\rangle\langle 11|\right)
\end{aligned}$$

Then the reduced density operator for the first particle, particle A, is:

$$\begin{aligned}\rho^A &= \mathrm{tr}_B(\rho) \\ &= \frac{1}{2}\left[\,\mathrm{tr}_B(|00\rangle\langle 00|) + \mathrm{tr}_B(|00\rangle\langle 11|) + \mathrm{tr}_B(|11\rangle\langle 00|) + \mathrm{tr}_B(|11\rangle\langle 11|)\,\right] \\ &= \frac{1}{2}\left[\,|0\rangle\langle 0|\,\mathrm{tr}_B(|0\rangle\langle 0|) + |0\rangle\langle 1|\,\mathrm{tr}_B(|0\rangle\langle 1|) + |1\rangle\langle 0|\,\mathrm{tr}_B(|1\rangle\langle 0|) + |1\rangle\langle 1|\,\mathrm{tr}_B(|1\rangle\langle 1|)\,\right] \\ &= \frac{1}{2}|0\rangle\langle 0| + \frac{1}{2}|1\rangle\langle 1|\end{aligned}$$

This result is consistent with the system A being in the state $|0\rangle$ with probability $\frac{1}{2}$ or state $|1\rangle$ with probability $\frac{1}{2}$, although other mixtures could produce the same result. Hence if we look at system A alone, it will be statistically indistinguishable from any of these mixtures. So we are free to think of it as being in *either* the state $|0\rangle$ *or* the state $|1\rangle$, with equal probability. In a sense this is not true, but it will make all the right predictions – until system A interacts with system B again, at which time the description could be revealed as false through unanticipated correlations between the two systems. This is how probabilities enter into this completely deterministic theory.

Discarding state

Suppose we have the two-qubit state $|\psi\rangle = a|00\rangle + b|11\rangle$, and we discard the second qubit. The reduced density operator for the first qubit can easily be shown to be:

$$\rho = |a|^2|0\rangle\langle 0| + |b|^2|1\rangle\langle 1|$$

So the first qubit will behave as if it were drawn from a mixture of the state $|0\rangle$ with probability $|a|^2$ or the state $|1\rangle$ with probability $|b|^2$. The only way in which it could behave differently is if it interacted with the discarded qubit, either directly or indirectly.

Now let us consider the effect that discarding a qubit has when it is a participant in a large entangled state. Suppose we have a system of $m+1$ qubits, and we plan on discarding the last one. We can write the state of this system in the following form:

$$|\psi\rangle = \sum_{x\in\{0,1\}^m} a(x)|x\rangle|g(x)\rangle$$

The notation $|x\rangle$ here implies that each of the first m qubits is in the state $|x_i\rangle$ with $x_i \in \{0,1\}$. If the $|g(x)\rangle$ term were omitted, then this equation would simply be the general form of the state of an m-qubit system, with $a(x)$ giving the amplitude associated with each basis state. Imagine this is the state we are truly interested in, but unfortunately it remains entangled with a "garbage" qubit left over as an intermediate step during a computation (the $|g(x)\rangle$ component). The density operator corresponding to the state $|\psi\rangle$ is then:

$$\begin{aligned}\rho &= |\psi\rangle\langle\psi| \\ &= \sum_x a(x)|x\rangle|g(x)\rangle \sum_y a^*(y)\langle g(y)|\langle y| \\ &= \sum_{x,y} a(x)a^*(y)\;|x\rangle\langle y|\;|g(x)\rangle\langle g(y)|\end{aligned}$$

Discarding the garbage qubit, the state of the remaining m qubits can be computed using the partial trace:

$$\begin{aligned}
\rho_A &= \mathrm{tr}_B(\rho) \\
&= \sum_{x,y} a(x)a^*(y)\ |x\rangle\langle y|\ \mathrm{tr}(|g(x)\rangle\langle g(y)|) \\
&= \sum_{x,y} a(x)a^*(y)\ |x\rangle\langle y|\ \langle g(y)|g(x)\rangle \\
&= \sum_{x,y} \left(a(x)a^*(y)\ |x\rangle\langle y|\ \frac{1}{2}\left(1 + (-1)^{g(x)+g(y)}\right)\right) \\
&= \frac{1}{2}\sum_{x,y} a(x)a^*(y)\ |x\rangle\langle y| + \frac{1}{2}\sum_{x,y}(-1)^{g(x)+g(y)}a(x)a^*(y)\ |x\rangle\langle y| \\
\rho_A &= \frac{1}{2}\sum_{x} a(x)|x\rangle \sum_{y} a^*(y)\langle y| + \frac{1}{2}\sum_{x}(-1)^{g(x)}a(x)|x\rangle \sum_{y}(-1)^{g(y)}a^*(y)\langle y| \\
&= \frac{1}{2}|\psi_{good}\rangle\langle\psi_{good}| + \frac{1}{2}|\psi_{bad}\rangle\langle\psi_{bad}|
\end{aligned}$$

where

$$\begin{aligned}
|\psi_{good}\rangle &= \sum_x a(x)|x\rangle \\
|\psi_{bad}\rangle &= \sum_x (-1)^{g(x)}a(x)|x\rangle
\end{aligned}$$

This density operator is consistent with the system being in either the state $|\psi_{good}\rangle$ or the state $|\psi_{bad}\rangle$ with equal probability. The state $|\psi_{good}\rangle$ is exactly the original state with the same superposition maintained but with the undesired qubit dropped. This is what we would like to get. The state $|\psi_{bad}\rangle$ is a superposition that is somewhat similar to the original, but the signs of the terms are affected by the qubit we are trying to discard. This state is rarely what we would like, unless the value of the qubit was constant. So we have only a 50% chance of successfully eliminating the effect of the undesired qubit, which will rarely be acceptable. Luckily, we will see in Section 3 that reversible logic permits a much more effective way of removing garbage bits by "uncomputing" them.

Measuring state

For historical reasons, the interpretation of the measurement process in quantum mechanics remains a sea of conflicting opinions and bad ideas. The mathematics of measurement is, in contrast, clear and unambiguous, so many of those involved in quantum computing prefer to simply forgo interpretation (Nielsen & Chuang 2001). But since this paper is about reversibility, and quantum measurement appears superficially to be probabilistic and irreversible, the mathematics of measurement cannot simply be taken at face value. A simple reversible interpretation is possible if we view measurement simply as a special case of losing state. Suppose we have a single qubit system as described earlier:

$$|\psi\rangle = a|0\rangle + b|1\rangle$$

We will see in later sections that it is a simple matter to entangle other systems with this one to get a state such as:

$$|\psi'\rangle = a|000\rangle + b|111\rangle$$

If the state of one of these qubits is not tracked further – perhaps it is in the uncontrolled environment – then the best local description of the system will be a mixed state:

$$\rho = aa^*|00\rangle\langle 00| + bb^*|11\rangle\langle 11|$$

This is consistent with being in the state $|00\rangle$ with probability $|a|^2$, and the state $|11\rangle$ with probability $|b|^2$. Of course, there is nothing inherently probabilistic or non-deterministic going on here – it just looks that way because of the untracked qubit.

If we imagine the second qubit to represent the output of a measuring device, we see that it is either $|0\rangle$ or $|1\rangle$. If the state $|\psi\rangle$ were $|0\rangle$, then the output would be $|0\rangle$ with probability 1. If the state $|\psi\rangle$ were $|1\rangle$, then the output would be $|1\rangle$ with probability 1. For all other states, there is some chance of reading a $|0\rangle$ and some chance of reading a $|1\rangle$. This analysis is consistent with the measurement postulate of quantum mechanics, which at first glance suggests an irreversible process but as shown here could be interpreted as a consequence of untracked correlated states.

Measurement can be generalized to multiple qubits and different choices of basis vectors. For the purposes of this paper, it is enough to consider measurement in the usual "computational basis", which is very straightforward. Suppose we have the following state:

$$|\psi\rangle = \frac{\sqrt{3}}{2}|000\rangle - \frac{1}{4}|001\rangle + \frac{1}{4}|100\rangle - \frac{1}{4}|101\rangle + \frac{1}{4}i|111\rangle$$

If we were to measure all the qubits, we would get one of the five basis states with probabilities equal to the squared amplitude of their coefficients. For example, the state $|000\rangle$ would be measured with probability $(\frac{\sqrt{3}}{2})^2 = \frac{3}{4}$, and all the others with probability $\frac{1}{16}$ each (regardless of phase). If instead we measured just the first qubit, we would get a $|1\rangle$ with probability equal to the sum of the probabilities associated with each of the terms consistent with that result – in this case $|100\rangle$, $|101\rangle$, and $|111\rangle$, with total probability $\frac{3}{16}$. The state of the system after the measurement would include just these three terms, renormalized to make the state a unit vector:

$$|\psi\rangle = \frac{1}{\sqrt{3}}|100\rangle - \frac{1}{\sqrt{3}}|101\rangle + \frac{1}{\sqrt{3}}i|111\rangle$$

Notice that the first qubit is effectively no longer part of the superposition and the state can be written as a product:

$$|\psi\rangle = |1\rangle \otimes (\frac{1}{\sqrt{3}}|00\rangle - \frac{1}{\sqrt{3}}|01\rangle + \frac{1}{\sqrt{3}}i|11\rangle)$$

The state after a measurement result of $|0\rangle$ can be derived in the same manner – it will be contain the $|000\rangle$ and $|001\rangle$ terms. This shows that an undesired qubit cannot be eliminated by measuring it, and we have already seen it is not a good idea to try just discarding it. The next section addresses the question this raises: is it possible to compute without erasing state information?

3 Reversible computing

As anyone foolish enough to place a laptop on their actual lap knows, contemporary computers generate a considerable amount of heat – even when every effort is made to minimize their energy consumption. We might hope that future technical developments will ameliorate this somewhat, and no doubt they will. But Landauer (1961) argued that a part of this heat generation is in fact unavoidable even in principle. He noted that whenever a bit in an unknown state is erased, there is a reduction in entropy within the state of the computer that must be matched by a corresponding increase in entropy elsewhere. The state of the bit is not in fact erased but is rather exported to an uncontrolled part of the computer, its thermal state.

The entropy associated with a system that is equally likely to be in any one of N states is $k \ln N$, where k is Boltzmann's constant. For a bit, the entropy is $k \ln 2$. If a process is applied to the bit to force it into a known state, the entropy associated with it becomes $k \ln 1 = 0$. If the bit is physically realized as part of a thermalized distribution, then energy will be dissipated in a heat flow of $dQ = TdS$, where T is the temperature and $dS = k \ln 2$ is the change in entropy (Gershenfeld 1996). This is about 3×10^{-21} Joules at room temperature. This insight – that erasing memory generates heat – was historically important in clearing up the status of Maxwell's demon. Maxwell's demon is an imaginary creature that guards a tiny door between two containers, opening and closing it strategically to allow selected molecules to pass and thereby arranging for the faster ("hot") molecules to be on one side and the slower ("cold") molecules on the other. This creates a temperature difference that could be used to do work, and so at first glance appears to violate the second law of thermodynamics. Putative "exorcisms" of the demon have been many and varied. Szilard (1929) published an influential paper that persuaded many that heat generated during measurement was the "savior" of the second law. In his paper Szilard noted in passing that the demon's *memory* of the result of a measurement had to be erased before it could close the thermodynamic cycle and continue on to measure the next particle. This insight was lost in confusion over the process of measurement until Bennett (1988) made the connection between the demon and Landauer's insight, showing that erasure of the result of a measurement was sufficient to save the second law.

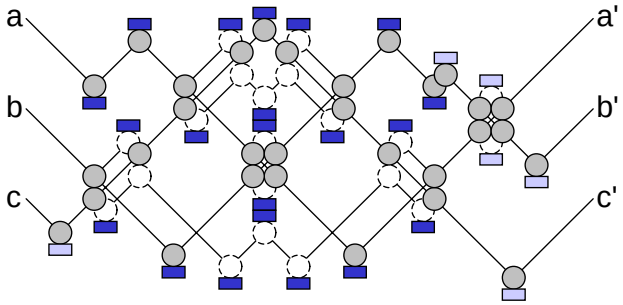
While exploring the consequences of the heat generated by erasing a bit, Landauer (1961) quickly realized that heat dissipation could be prevented by simply storing all bits that would otherwise be erased. But this was simply a delaying action; these "garbage" bits would accumulate continuously over the course of a computation. Landauer assumed that the programmer would eventually have no choice but to erase the bits so that storage space could be reused – at which point heat would be dissipated. And so he concluded that computing was necessarily an irreversible process. But Bennett (1973) showed that this conclusion was premature, and that it is in fact possible to compute in an entirely reversible fashion. He demonstrated this by constructing a reversible Turing machine. The machine first performed the desired computation, saving all necessary state to preserve reversibility at each state transition, essentially as Landauer envisioned. Then the machine ran in reverse, step by step undoing everything it had computed, and so succeeds in returning the machine's memory back to its original state without having to forcibly reset the state of any bits. In fact unknown to Bennett this much of his construction had been anticipated earlier by Lecerf (1963) in a paper concerned with a problem in the "diagonalization of homomorphisms of free monoids". Lecerf was apparently not aware of Landauer's work and never made (or wanted to make) the connection to heat generation.

But of course the machine described so far is useless since it uncomputes the desired output of the computation along with everything else. Bennett's innovation over Lecerf was to add an intermediate phase, before the machine ran in reverse, where the desired output was copied to an initially blank auxiliary tape. This simple addition meant that at the end of the process, the desired output was available while all intermediate results generated along the way were completely erased.

Unfortunately the space needed to store all the intermediate state information required by Bennett's construction can be quite large; it is proportional to the length of time the machine runs for, which could be exponential in the size of the storage needed by an irreversible equivalent. Bennett pointed out that if a computation can be broken down into sequential modules, each module can be individually reversed and so space can be reused as the computation proceeds. In Bennett (1989), he develops this argument to show that an irreversible Turing machine using time T and space S can be simulated by a reversible Turing machine requiring time $O(T^{1+\varepsilon})$ and space $O(S \ln T)$ for $\varepsilon > 0$. Levine & Sherman (1990) quickly pointed out that these bounds are somewhat misleading, because of a hidden factor that diverges as ε approaches 0. Their refined bounds show the reversible machine running in time $\Theta(T^{1+\varepsilon}/S^{\varepsilon})$ and space $\Theta(S(1 + \ln(T/S)))$. This is the best known current result for trading off time and space for a reversible simulation of an irreversible machine. Conveniently, it preserves the complexity classes P and NP, although other classes are not preserved. At anoth-

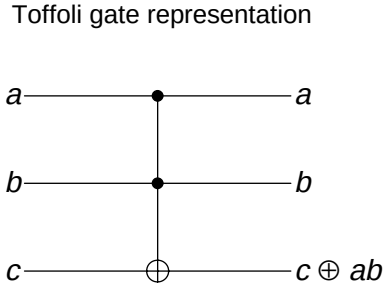
er extreme, Lange, McKenzie & Tapp (1997) have shown that an irreversible Turing machine can be simulated reversibly in linear space, but unfortunately their construction uses exponential time.

Further bolstering the idea that computation need not cause an increase in entropy, Edward Fredkin developed an ingenious reversible logic gate that could in principle be implemented as billiard balls bouncing off perfectly elastic walls (see Figure 2). This gate, called the Fredkin gate, is universal for digital logic and similar techniques to those of Bennett can be used to eliminate intermediate results (Fredkin & Toffoli 1982). In logical terms, the Fredkin gate is a controlled-swap operation. It has three inputs and three outputs. The three signals pass through unchanged, except when the a control input is active; in that case the other two signals are switched. Fredkin's collaborator, Toffoli, also developed a universal reversible logic gate that is easier to work with in practice but doesn't have a similar implementation in billiard-ball physics (see Figure 3).



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	1	0	1
1	0	0	1	0	0
1	0	1	0	1	1
1	1	0	1	1	0
1	1	1	1	1	1

Fig. 2. A billiard ball implementation of a Fredkin gate, following Nielsen & Chuang (2001). Rectangles represent walls, circles represent possible ball positions. If no ball enters at c, the "control" input, then balls entering at a and b bounce through to a' and b' respectively. If a ball does enter at c, then if balls enter at either a or b they will exit from the opposite outputs, b' or a'.



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

Fig. 3. The Toffoli gate. Now a and b are control inputs and pass through unchanged; if both are true, then a target input c is flipped. The Toffoli gate is universal and reversible – in fact it is its own inverse, since flipping the target input twice will leave it unchanged.

It is now time to return to the quantum world to see how reversible logic can be mapped onto the evolution of a quantum system.

4 Quantum Evolution

In Section 2 we saw that erasing qubits in a quantum computer could scramble the state of the machine. Then Section 3 showed that it is in fact possible to perform useful computations without discarding state in this way. We now examine how quantum system evolve, and how to match that evolution to reversible logic.

An isolated quantum system evolves in time according to Schrödinger's equation:

$$i\hbar\frac{d|\psi\rangle}{dt} = H|\psi\rangle$$

where H is a fixed Hermitian operator called the *Hamiltonian* of the system, and $\hbar$ is Planck's constant (divided by 2π). If we consider the evolution of the system in some period t_1 to t_2 from an initial state $|\psi\rangle$, the final state will be:

$$|\psi'\rangle = U|\psi\rangle$$

where U is a unitary operator that depends only on the times t_1 and t_2 and which can easily be derived from the Hamiltonian for the system. In quantum computing, the system Hamiltonian is manipulated to yield a desired unitary operation on the quantum state – in general the system will not in fact be isolated as claimed above, but strongly driven to yield a particular Hamiltonian. When dealing with mixed states, it is easier to express the system evolution in terms of the density operator representation introduced in Section 2. The evolution of the density operator of the system in some period t_1 to t_2 from an initial state ρ is given by:

$$\rho' = U\rho U^\dagger$$

where U is the same unitary operator as before and $U^\dagger$ is the adjoint $(U^*)^T$. In the case of unitary operators, $U^\dagger$ is in fact the inverse of U. This means that the inverse always exists, and so we can conclude that the evolution of a quantum system is reversible.

Composing unitary operators

A unitary operator applied to an n-qubit system can be described by an $n \times n$ complex-valued matrix whose eigenvalues $e^{i\theta_k}$ are all of modulus one. The space of possible operators is very large. It contains all reversible logic operations as a small subset, and so a quantum computer is at least as powerful as a classical reversible computer. But there are also many operators with no classical equivalent. For example, while there are only two reversible logic operations that can be applied to a bit – the identity and NOT – there are a continuum of operators that can be applied to a qubit. An important one-qubit operator is the Hadamard operator, written H (forgetting now about Hamiltonians):

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

This matrix representation assumes that the state of the qubit is written as a column vector:

$$a|0\rangle + b|1\rangle = a\begin{bmatrix} 1 \\ 0 \end{bmatrix} + b\begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} a \\ b \end{bmatrix}$$

Applying the Hadamard operator to the state $|0\rangle$ gives a non-classical result familiar from Section 2:

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

Another useful family of operators lets us change the phase of the $|0\rangle$ and $|1\rangle$ components relative to each other. Examples include the S and T operators, called the *phase* and $\pi/8$ operators respectively for historical reasons:

$$S = \begin{bmatrix}1 & 0\\ 0 & i\end{bmatrix} \qquad T = \begin{bmatrix}1 & 0\\ 0 & e^{i\pi/4}\end{bmatrix}$$

For example, S applied to the state $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ gives $\frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle)$. Manipulating relative phases is vital for controlling interference effects.

If we have two qubits, an important operator is controlled-NOT, or $CNOT$. If we assume states are expressed as $a|00\rangle + b|01\rangle + c|10\rangle + d|11\rangle = (a, b, c, d)^T$, then $CNOT$ corresponds to the following matrix:

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

Its effect is to swap terms in $|10\rangle$ and $|11\rangle$. In other words, it flips the logical state of the second ("target") qubit if the first ("control") qubit is $|1\rangle$, and leaves it alone otherwise:

$$\begin{aligned} a|00\rangle + b|01\rangle + c|10\rangle + d|11\rangle &\overset{\text{CNOT}}{\longmapsto} a|00\rangle + b|01\rangle + c|11\rangle + d|10\rangle \\ &= \; a|00\rangle + b|01\rangle + d|10\rangle + c|11\rangle \end{aligned}$$

Since quantum evolution is linear, it is always possible to compute the effect of multi-qubit operators on complicated states by applying the operator to each basis vector individually, then scaling and summing the result.

It turns out that three of the operators mentioned so far – Hadamard (H), $\pi/8$ (T), and controlled-NOT ($CNOT$) can together be composed to form an arbitrarily precise approximation to any unitary operator on n qubits (Boykin, Mor, Pulver, Roychowdhury & Vatan 1999). This composition is analogous to building combinatorial digital circuits out of a universal set of gates. That these operators are universal is somewhat surprising, since classically at least one three-input gate such as the Toffoli gate is needed for universality in reversible logic. But because of the extra types of operations that exist in a quantum setting, it is in fact possible to construct the Toffoli gate out of one-qubit and two-qubit operators. The next section examines universality in more detail.

5 Universal quantum gates

Constructing a desired unitary operator by composing a set of "building block" operators that apply to a small number of qubits at a time is analogous to conventional digital circuit design. Deutsch (1989) introduced a representation called a quantum gate array which consists of a set of "wires" (corresponding to qubits) punctuated by "gates" (corresponding to unitary operators). Each gate take a certain number of inputs, applies their associated operator, and presents the result on the same number of outputs. For example Figure 4 shows

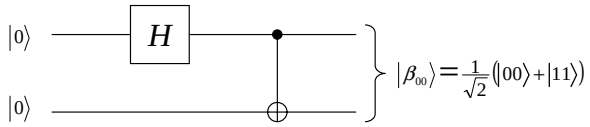


Fig. 4. An example of a simple gate array, or quantum circuit. The horizontal lines represent qubits. Time flows from left to right. The box marked H applies a Hadamard operator to the top qubit. The vertical line terminating in a cross represents the $CNOT$ operator.

a simple gate array involving the operators H and $CNOT$. Horizontal lines in this circuit represent individual qubits, with their initial values shown on the left. Gates are often represented as simple labelled boxes, as for the Hadamard gate shown (the H acting on the uppermost qubit). The controlled-NOT gate is drawn as a vertical line beginning from the control qubit and terminating on the target qubit with an "XOR" symbol ($\oplus$). The activity of the gate array is as follows. The input state is $|0\rangle \otimes |0\rangle = |00\rangle$. Then a Hadamard operator is applied to the first qubit, leaving the second unchanged, giving:

$$\begin{aligned}\big(H|0\rangle\big) \otimes |0\rangle &= \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |0\rangle \\ &= \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|10\rangle\end{aligned}$$

We now apply a controlled-NOT, with the first qubit as the control and the second as the target. This swaps terms in $|10\rangle$ for terms in $|11\rangle$. So the result is the state:

$$\frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle$$

which is the state $|\beta_{00}\rangle$ introduced in Section 2 as an example of a state that cannot be written as a product of states of its component qubits. So this simple circuit can take the entirely classical state $|00\rangle$ and produce an entangled state.

A set of quantum gates is universal for quantum computation if they can be combined to approximate any unitary operator to any degree of accuracy. There are many possible choices for such a set. Barenco et al. (1995) showed that the set of all single-input quantum gates, augmented with the controlled-NOT gate, is universal. In fact this set can be used to construct all unitary operators *exactly* – no approximation is needed. This draws on the work of Deutsch (1989) who showed that a generalization of the Toffoli gate is universal for quantum computation. The Toffoli gate applies a bit-flip to its target when the control inputs are both $|1\rangle$; Deutsch's generalization allows any one-qubit operator to be applied the target in place of a bit-flip, and permits any number of control inputs. Barenco et al. (1995) give a decomposition for these "controlled-U" gates into gates with fewer inputs. For example, Figure 5 shows how a three-qubit controlled-U gate can be constructed from operations over at most two qubits. Constructions of this nature demonstrate that two-qubit gates are sufficient for universality. Barenco et al. (1995) also show how a controlled-U gate with a single control input can be implemented by the controlled-NOT gate and single-qubit gates, showing that universality is possible with just a single type of two-qubit gate. This result hinges on the fact that any single-qubit unitary operator U can be decomposed into the following form:

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

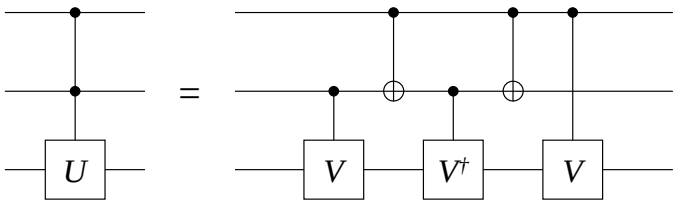


Fig. 5. Implementing a controlled-U gate with two control inputs using gates with single control inputs. V is chosen such that $V^2 = U$. If the two control qubits are $|1\rangle$ then the transformation applied to the target is $VV = U$. If either of the control qubits is $|0\rangle$ then the transformation applied is either I or $VV^\dagger = I$. Hence this construction applies the correct control logic.

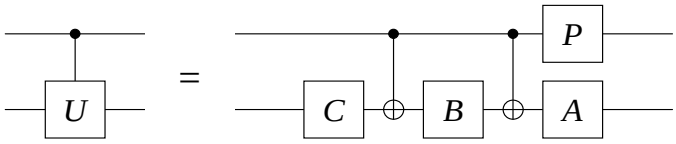


Fig. 6. Implementing a controlled-U gate with controlled-NOT gates and single qubit gates. The gates A, B, and C are as described in the text – ignore the P gate for now. If the control qubit (the upper line) is $|0\rangle$, then the target qubit (lower) is transformed by $ABC = I$. If the control qubit is $|1\rangle$, then the target qubit is transformed by $AXBXC = e^{-i\alpha}U$, which is the desired operator up to a phase. The P gate cancels this phase by applying $\begin{pmatrix} 1 & 0 \\ 0 & e^{-i\alpha} \end{pmatrix}$.

It is then possible to construct three operators A, B, and C such that $ABC = I$ and $e^{i\alpha}AXBXC = U$, where X is the bit-flip operator.

$$A = \begin{pmatrix} e^{-i\frac{\beta}{2}} & 0 \\ 0 & e^{i\frac{\beta}{2}} \end{pmatrix} \begin{pmatrix} \cos\frac{\gamma}{4} & -\sin\frac{\gamma}{4} \\ \sin\frac{\gamma}{4} & \cos\frac{\gamma}{4} \end{pmatrix} \quad B = \begin{pmatrix} \cos\frac{\gamma}{4} & \sin\frac{\gamma}{4} \\ -\sin\frac{\gamma}{4} & \cos\frac{\gamma}{4} \end{pmatrix} \begin{pmatrix} e^{i\frac{\beta+\delta}{4}} & 0 \\ 0 & e^{-i\frac{\beta+\delta}{4}} \end{pmatrix} \quad C = \begin{pmatrix} e^{i\frac{\beta-\delta}{4}} & 0 \\ 0 & e^{-i\frac{\beta-\delta}{4}} \end{pmatrix}$$

This decomposition is all that is needed to build up the controlled-U gate from controlled-NOT gates and single qubit gates, as shown in Figure 6. The constructions so far require the ability to perform arbitrary single-qubit operators. If we are willing to settle for approximations to the desired result, it is possible to use just a small set of basic single-qubit operators, for example Hadamard and $\pi/8$ gates (Boykin et al. 1999). This simulation is efficient – if n is the number of gates in the original circuit, the number of gates in the approximation is polynomial in $\log(n/\varepsilon)$ where ε is the accuracy of the simulation.

In practice, implementation details will dictate which operators are easy to perform and which are harder. It is easy to leverage a small set of operators into a much larger set by combining the Hamiltonians that generate them. For example it is possible to enact the Hamiltonian $A + B$ given the ability to enact A and B, by switching rapidly between them. Various versions of the Trotter formula provide the necessary recipe – for example:

$$e^{i(A+B)\Delta t} = e^{iA\Delta t/2}e^{iB\Delta t}e^{iA\Delta t/2} + O(\Delta t^3)$$

This approximation implies that by switching at a sufficiently small time step Δt, we can enact the Hamiltonian $A + B$ by enacting A for $\Delta t/2$, then B for Δt, then A for $\Delta t/2$. Another very productive formula is that of Baker-Campbell-Hausdorf:

$$e^{(A+B)\Delta t} = e^{A\Delta t}e^{B\Delta t}e^{-\frac{1}{2}[A,B]\Delta t^2} + O(\Delta t^3)$$

This can be arranged to show how to enact $i[A, B]$ in terms of A, B, and $A + B$. So if we know how to implement two Hamiltonians, we can implement their commutator. Similarly, we can generate $[A, [A, B]]$ etc. Unless A and B are very unfortunately chosen, we can in fact generate any Hamiltonian this way, in principle (Lloyd 1995).

But a crucial issue not touched on at all yet is whether gates can be implemented in a fault tolerant manner, so that errors do not accumulate during a computation. This problem is addressed in the next two sections.

6 Error correction

In digital circuitry, logical states are represented by extremes of a continuous physical parameter. If a signal starts to deviate away from its nominal value, it can easily be pushed back long before it might be interpreted erroneously as corresponding to another state. This ceaseless "restandardization" is missing from analog computers. When state information is stored densely in a continuum, then a slight drift in a signal may be enough to reach the nominal value for a different state – and so the drift cannot be detected and corrected. A worrying early criticism of quantum computing was that it seemed to be essentially analog in nature, and so restandardization or error correction would not be possible (Preskill 1998a). If this were true, then constructing a large-scale quantum computer would be an impossible dream. It is certainly true that the state of a set of qubits lies in a continuous Hilbert space, where each point on the unit hypersphere in that space corresponds to a perfectly valid state. Even worse, the variety of errors that can occur in a quantum setting is far greater than in classical scenarios. We have already seen that if just a single degree of freedom in the environment becomes entangled with the state of the machine, its ability to perform useful computation may well be destroyed. Achieving error correction in a quantum setting is further hindered by the fact that an incautious measurement can effectively destroy the state we are trying to preserve. All very daunting indeed.

The first crucial step in quantum error correction is to demonstrate that an error can be detected and corrected without learning anything about the state being protected – since we know we cannot measure that state without disturbing it (Nielsen & Chuang 2001). Consider one of the simplest classical codes, where a bit is simply repeated three times:

$$|0\rangle \longmapsto |000\rangle$$
$$|1\rangle \longmapsto |111\rangle$$

The only error that can occur classically is a bit flipping from $|0\rangle$ to $|1\rangle$ or vice versa, and with this code we can correct for up to one such error by taking a majority vote. In a quantum setting, we need to consider what happens to non-classical states:

$$a|0\rangle + b|1\rangle \longmapsto a|000\rangle + b|111\rangle$$

If we measure these three qubits to determine if there has been an error, we will destroy the very state information we wish to protect. But for this code we can safely measure parity information. To see this, consider a bit-flip on the first qubit. This will result in:

$$a|000\rangle + b|111\rangle \longmapsto a|100\rangle + b|011\rangle$$

Regardless of what the state we are protecting is, the encoded state has even parity before the error and odd parity afterwards. The circuit shown in Figure 7 shows one way to measure that parity without learning anything about the protected state. We can easily use the parity to diagnose whether a (single) error has occurred. To determine which qubit has been flipped, we can measure the parity of each pair of qubits. The two unflipped qubits will have even parity as a pair, identifying the remaining qubit is the erroneous one. We can then apply

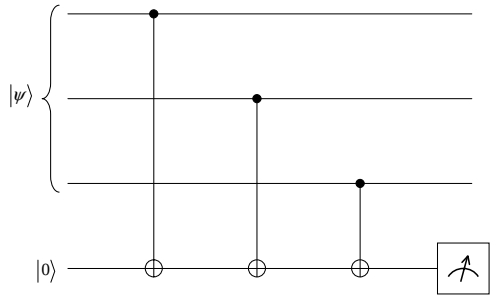


Fig. 7. Measuring parity using an ancillary qubit (a spare qubit in a known state is often called an ancilla). The ancilla is flipped once for each qubit in the state $|1\rangle$, so its final state reflects the overall parity.

a bit flip operation to correct that qubit, which is a reversible operation, without ever learning anything about the protected state.

But this scheme is incomplete, since bit-flips are not the only errors that can occur in a quantum system. Phase-flip errors are also possible. If we wish to protect against this kind of error, forgetting about bit-flips for the moment, we can use a different but analogous code:

$$|0\rangle \longmapsto \frac{1}{2\sqrt{2}}(|0\rangle + |1\rangle)(|0\rangle + |1\rangle)(|0\rangle + |1\rangle)$$
$$|1\rangle \longmapsto \frac{1}{2\sqrt{2}}(|0\rangle - |1\rangle)(|0\rangle - |1\rangle)(|0\rangle - |1\rangle)$$

Instead of measuring parity, we now measure products of phases, but otherwise the analysis is the same as before. Now suppose we wish to protect against either a bit-flip or a phase-flip. Then we can simply chain the two codes given above to get the following 9-qubit code:

$$|0\rangle \longmapsto \frac{1}{2\sqrt{2}}(|000\rangle + |111\rangle)(|000\rangle + |111\rangle)(|000\rangle + |111\rangle)$$
$$|1\rangle \longmapsto \frac{1}{2\sqrt{2}}(|000\rangle - |111\rangle)(|000\rangle - |111\rangle)(|000\rangle - |111\rangle)$$

This is called the Shor code (Shor 1995). We first correct for bit flips within each triplet of qubits. A phase flip of any qubit within a triplet has the same overall effect, so we can correct phases just as before.

Through a stroke of serendipity, it turns out that correcting for bit flips and phase flips corrects for the whole continuum of errors that can affect a qubit – from a tiny drift, to complete replacement with a random state. Measuring whether either of these two types of errors occurred actually discretizes the error modes, which is what finally saves quantum computing from the analog curse.

7 Fault tolerance

Error correction is useful for transmitting and storing quantum states. But to compute with the state, it must be possible to apply operators to it. If the state needs to be decoded to do this, then it will be vulnerable to corruption. So *fault tolerant* mechanisms have been developed that allow operators to be applied directly to the encoded form of a quantum state. These mechanisms need to be carefully designed so that they do not propagate errors excessively. How difficult this is to achieve depends on the coding method used. The 9-qubit

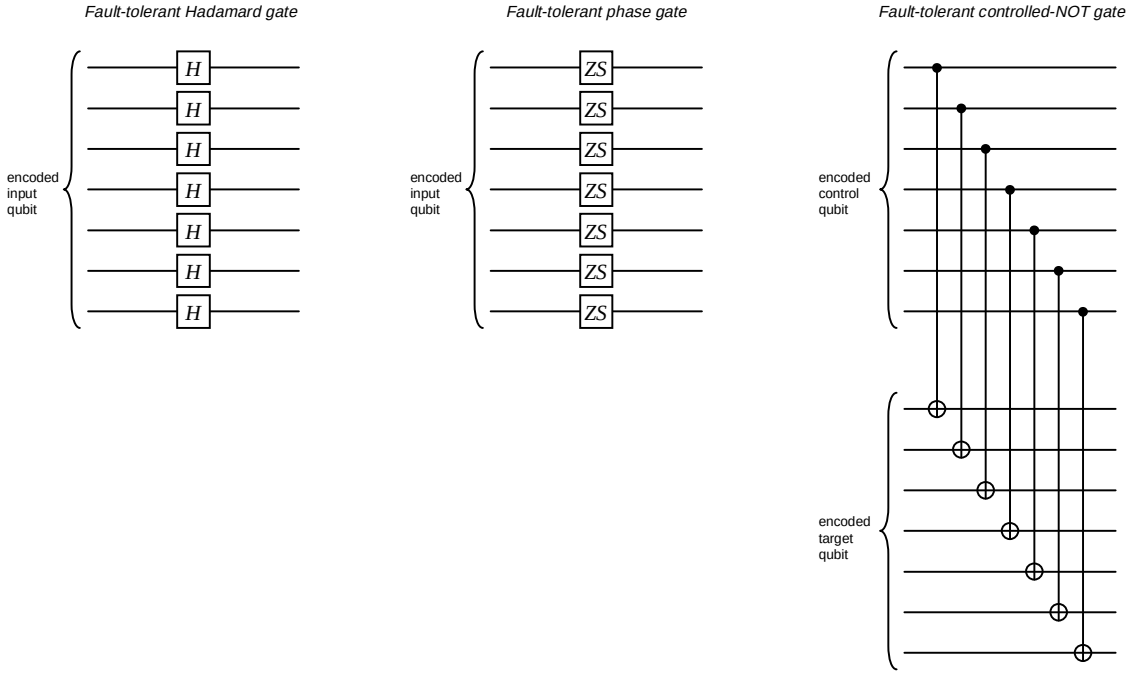


Fig. 8. Fault tolerant implementations of the Hadamard (left), phase (center) and controlled-NOT gates (right), for use with the Steane code. This style of implementation, where the same operation is applied to each of the qubits in a code block independently is called "transversal" and is desirable because it avoids propagating single errors to multiple qubits within the same code block.

Shor code given earlier is easy to understand but is not ideal for implementing fault tolerance. The Steane code is a popular 7-qubit code that also can protect against a one-qubit error (Steane 1996). It is less intuitive than the Shor code, but does permit remarkably simple fault-tolerant gates to be constructed. For completeness, the code is as follows:

$$
\begin{aligned}
|0\rangle \longmapsto \tfrac{1}{\sqrt{8}} \big[&|0000000\rangle + |1010101\rangle + |0110011\rangle + |1100110\rangle + \\
&|0001111\rangle + |1011010\rangle + |0111100\rangle + |1101001\rangle \big] \\
|1\rangle \longmapsto \tfrac{1}{\sqrt{8}} \big[&|1111111\rangle + |0101010\rangle + |1001100\rangle + |0011001\rangle + \\
&|1110000\rangle + |0100101\rangle + |1000011\rangle + |0010110\rangle \big]
\end{aligned}
$$

For this code, fault-tolerant versions of the Hadamard and controlled-NOT gates can be constructed very simply, as shown in Figure 8. The fault tolerant Hadamard is built by applying a normal Hadamard gate separately to each of the seven qubits in an encoded block representing a single logical qubit. If one of the gates fails, that error affects one qubit within the block, which can be corrected. The fault tolerant controlled-NOT gate applies a controlled-NOT to corresponding qubits within the blocks for the logical control and target qubits. A failure in one of those gates will affect no more than one qubit within each block, which can be corrected.

The Hadamard, phase, and controlled-NOT gates together are not universal, but they are sufficient to construct many circuits that are important for communication and coding. These gates belong to the Clifford group, a group that plays a key role in the theory of quantum error correction and fault tolerance.

The Clifford group

It is possible to analyze the behavior of quantum circuits in terms of how operators evolve, as opposed to how the state itself evolves. In other words, instead of characterizing a gate by how it transforms the basis states, we might describe how it transforms operators on the original state. For example, if we decide to track the effect applying the gate U has on the operator H, we can use the following identity:

$$UH|\psi\rangle = UHU^\dagger U|\psi\rangle$$

Remember that $U^\dagger U = I$ since U is unitary. This identity implies that after U is applied, the operator $UHU^\dagger$ has the same affect as applying H before U. So in a sense U transforms H to $UHU^\dagger U$. With a careful choice of operators and notation, this "Heisenberg" representation can give a very terse, compact description of the operation of many important quantum circuits (Gottesman 1999). A good set of operators to choose are tensor products of the Pauli operators, namely:

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

I is the usual identity operator. The X operator flips the $|0\rangle$ and $|1\rangle$ components of a state; it is sometimes called the "bit-flip" operator. The Z operator switches the phase of the $|1\rangle$ component while leaving the $|0\rangle$ component unchanged, and is often called the "phase-flip" operator. The Y operator is just the product of X and Z along with a global phase. The Pauli operators together form a basis for unitary matrices on a single qubit. Tensor products of the Pauli operators form a basis for arbitrary unitary matrices.

The Clifford group is the set of unitary operators which, when composed with a tensor product of Pauli operators, yields another tensor product of Pauli operators (up to a global phase of ± 1 or $\pm i$). The Hadamard operator has this property. So does controlled-NOT. And so do all the Pauli operators themselves. By staying within the set of Pauli operators, compositions of operators within the Clifford group can be expressed concisely in the Heisenberg implementation. And by considering states that are left unchanged by a set of operators, codes such as the Steane code find a much simpler and tractable representation.

An impressive body of work has been developed for analyzing and constructing circuits from gates within the Clifford group (Gottesman 1999). Unfortunately operators such as the $\pi/8$ gate do not map Pauli operators to Pauli operators, and so are not amenable to these techniques. So universal quantum computation cannot be handled within this framework. One proposal for a systematic procedure for implementing gates outside of the Clifford group finds its roots in quantum teleportation.

8 Quantum teleportation

A quantum state cannot in general be duplicated, unless it is known to be drawn from one of a set of mutually orthogonal possibilities (such as the classical $|0\rangle$ and $|1\rangle$ states). This is called the no-cloning theorem (Wootters & Zurek 1982). When first discovered, this result was thought to rule out even the theoretical possibility of one day developing "Star Trek" style teleportation, since the quantum state of matter could not be copied. But it was later discovered that despite this limitation, quantum state can in fact be transmitted perfectly across a classical communications channel if the sender and receiver share certain resources (Bennett, Brassard, Crépeau, Jozsa, Peres & Wootters 1993). This cannot be used to clone a state because the original

Fig. 9. No cloning sign (left) adapted from a lecture by Peter Shor. The fact that quantum states cannot be copied prompted early skepticism about the utility of quantum information processing. As well as casting doubt on teleportation, it had the more serious implication that quantum error correction and long distance communication might be impossible. Thankfully this turned out not to be a serious detriment, and now quantum coding is a thriving cottage industry turning out hundreds of papers about Alice and Bob (right), the mythical characters whose convoluted public and private lives make them insatiable consumers of all the latest coding technology. (Cartoon created by John Richardson for Physics World, March 1998).

is unavoidably destroyed in the process. This method is called teleportation despite the fact that it involves transmission of state as opposed to matter.

Quantum teleportation is usually explained with the aid of two characters Alice and Bob, famous in coding theory for wishing to communicate with each other under ludicrously hostile conditions. Suppose Alice and Bob each possess one qubit of the entangled pair of qubits $|\beta_{00}\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$. Then Alice can pass the state of a qubit $|\psi\rangle = a|0\rangle + b|1\rangle$ to Bob by carefully interacting $|\psi\rangle$ with her side of the entangled pair, and sending him some classical information. As a result of this procedure, Bob ends up with a bit-flipped and/or phase-flipped variant of $|\psi\rangle$: either $|\psi'\rangle = a|0\rangle \pm b|1\rangle$ or $|\psi'\rangle = b|0\rangle \pm a|1\rangle$. The classical information Alice sends to Bob encodes which of these four variants Bob will have, which occur with equal probability. Once Bob receives this information, he can easily apply the appropriate transformation to convert $|\psi'\rangle$ into the desired state $|\psi\rangle$. But without this information, Bob can learn absolutely nothing about the state $|\psi\rangle$ – since a mixture of the four possibilities above give a density matrix that is independent of $|\psi\rangle$. This fact is important since otherwise the procedure would permit information to be communicated faster than the speed of light, leading to causal inconsistencies and grandfather-cide.

Figure 10 shows a circuit for performing teleportation of a single qubit. Let $|\psi\rangle = a|0\rangle + b|1\rangle$ be the state to be teleported. Then the initial state of the system is:

$$\begin{aligned} |\Psi\rangle &= |\psi\rangle|\beta_{00}\rangle \\ &= (a|0\rangle + b|1\rangle)\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \\ &= \frac{1}{\sqrt{2}}(a|000\rangle + a|011\rangle + b|100\rangle + b|111\rangle) \end{aligned}$$

Applying the controlled-NOT operator, we flip the second qubit whenever the first is $|1\rangle$:

$$|\Psi'\rangle = \frac{1}{\sqrt{2}}(a|000\rangle + a|011\rangle + b|110\rangle + b|101\rangle)$$

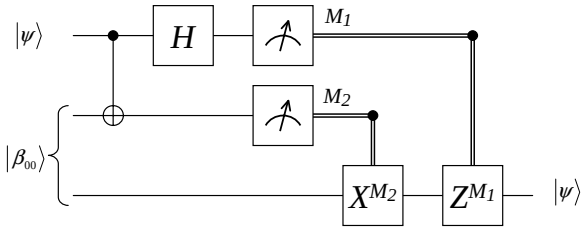


Fig. 10. Gate array for performing quantum teleportation of a single qubit.

Applying the Hadamard gate to the first qubit maps $|0\rangle$ to $|+\rangle$ and $|1\rangle$ to $|-\rangle$:

$$\begin{aligned}
|\Psi''\rangle &= \frac{1}{2}\left[\,(|0\rangle+|1\rangle)\otimes a|00\rangle + (|0\rangle+|1\rangle)\otimes a|11\rangle + (|0\rangle-|1\rangle)\otimes b|10\rangle + (|0\rangle-|1\rangle)\otimes b|01\rangle\,\right] \\
&= \frac{1}{2}(a|000\rangle + a|100\rangle + a|011\rangle + a|111\rangle + b|010\rangle - b|110\rangle + b|001\rangle - b|101\rangle) \\
&= \frac{1}{2}\left[\,|00\rangle(a|0\rangle+b|1\rangle) + |01\rangle(b|0\rangle+a|1\rangle) + |10\rangle(a|0\rangle-b|1\rangle) + |11\rangle(-b|0\rangle+a|1\rangle)\,\right]
\end{aligned}$$

If we were now to discard the first two qubits, the remaining qubit is in the state $|\psi\rangle$ up to a bit-flip and/or a phase-flip. If we instead measure the first two qubits – represented by the meter symbols in the figure – we learn nothing about $|\psi\rangle$ but we do learn how to recover it from the third qubit. If we measure the first two qubits as $|00\rangle$ then the third is exactly $|\psi\rangle$. If we measure $|01\rangle$, then we need to apply a bit-flip operator (X) to recover $|\psi\rangle$ – the double lines in the figure represent the classical control needed to do this. Measuring $|10\rangle$ tells us we need to apply a phase-flip operator (Z). And for $|11\rangle$ we need to do both a bit-flip and phase-flip. So in each of the four cases we can recover $|\psi\rangle$ without ever learning anything about its state in the communication process.

The measurement step in Figure 10 is important for teleportation as its results can be sent through a classical communication channel, so that no quantum interactions are needed between Alice and Bob after the initial distribution of the entangled pair $|\beta_{00}\rangle$. We will be concerned with using teleportation machinery for gate construction, following Gottesman & Chuang (1999), so we are free to postpone the measurement step, replacing classical control with quantum control as shown in Figure 11. Delaying measurement this way never changes the operation of a quantum circuit. If we do so, then the final state of the system is:

$$|\Psi'''\rangle = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)\otimes(a|0\rangle + b|1\rangle)$$

And we can clearly just discard the first two qubits – there is in fact no need to measure them, since the result no longer affects the third qubit.

9 Teleportation for gate construction

Section 7 showed one way to implement fault-tolerant versions of Hadamard and controlled-NOT gates, both of which are members of the Clifford group and have special properties that make them easy to work with. But we need at least one more gate drawn from outside the Clifford group for universal fault-tolerant computation. For example, if the $\pi/8$ gate also had a fault-tolerant implementation, then all operators could be implemented in a fault-tolerant manner. Gottesman & Chuang (1999) developed a procedure for doing this, using the machinery of quantum teleportation. The teleportation process described in Section 8 depends on the availability of an ancillary state $|\beta_{00}\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$. If this state is different, the outcome of the teleportation circuit

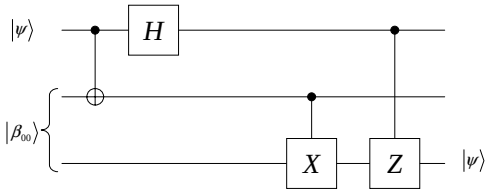


Fig. 11. Using teleportation without measurement. If classical control is replaced with quantum control, we can simply discard the qubits that were previously measured. Implementations of teleportation have used this approach, simply letting the discarded qubits decohere rather than explicitly measuring them (Nielsen et al. 1998).

may be different. The basic question addressed by Gottesman & Chuang (1999) is whether by careful choice of the ancilla, the teleportation procedure can be made to implement a useful transformation on its input. And in fact it can. With the right choice of ancilla, it can implement gates such as the $\pi/8$ gate. Since the gates in the quantum circuit itself are all in the Clifford group, that implies that the $\pi/8$ gate can be implemented in a fault-tolerant manner – assuming the ancilla itself can be prepared reliably.

Figure 12 demonstrates the technique in action. Since we are no longer interested in teleportation per se, we can permit the "sender" and "receiver" to interact quantum mechanically, allowing the circuit to be considerably simplified (Zhou, Leung & Chuang 2000). In fact we can eliminate one of the ancillary qubits used in the general teleportation circuit. To derive a circuit that applies the $\pi/8$ gate, we simply apply that gate to the output of the teleportation circuit and then commute it backwards towards the inputs. In the end, we can arrange the circuit so that the $\pi/8$ gate applies only to the ancilla. The remainder of the circuit can be produced using operators in the Clifford group, and so can be implemented in a fault tolerant manner. We can produce $TH|0\rangle$ in a fault tolerant manner by simple quality control – generate the state by non-fault-tolerant means, then measure it to ensure it has the correct value, and simply start over if it doesn't. Of course the success of this procedure depends on being able to perform measurement in a fault tolerant way, which luckily is possible (Shor 1996).

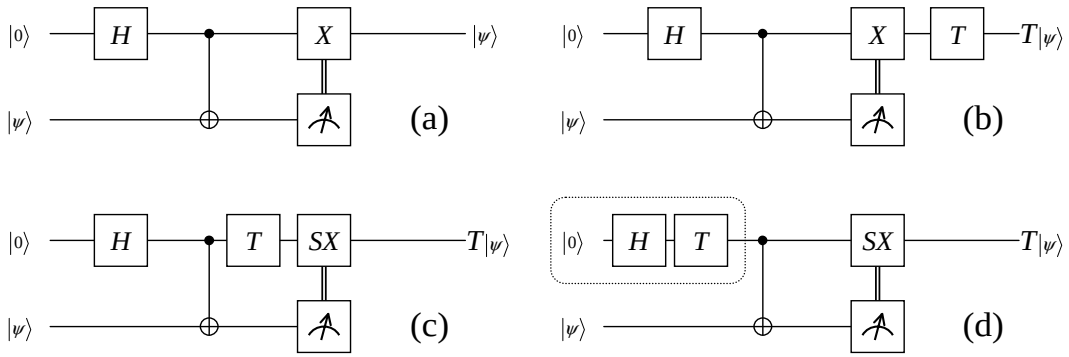


Fig. 12. An implementation of the $\pi/8$ (T) gate that is compatible with fault tolerance. This follows a construction in (Zhou et al. 2000). In (a), a simple circuit is shown that transfers an unknown state $|\psi\rangle$ from one qubit to another. In (b), we apply a T gate to the output to get the desired result $T|\psi\rangle$. Now we commute T back through the circuit, making any changes necessary to other gates as we go. In (d) the gate is now being applied to a constant input, and so its result can be prepared independent of the main computation. All the gates outside of the dotted line have fault-tolerant implementations.

The procedures introduced in Gottesman & Chuang (1999) and developed in (Zhou et al. 2000) allow direct fault tolerant implementations of any operator U with the property that for any Pauli operator P, the operator $UPU^\dagger$ is in the Clifford group. This covers important gates such as the $\pi/8$ gate as we've already seen, but also the Toffoli gate which can be implemented using a two-input generalization of the teleportation circuit.

10 Conclusions

Information cannot be destroyed in a reversible universe, but it can be dispersed and made inaccessible. This is what occurs in a classical computer every time it performs a nominally irreversible operation. But quantum computers can't disperse state information without forfeiting the ability to control entanglement, which is the only reason to build a quantum computer in the first place. So while reversibility is merely optional for classical computation, it is a requirement for quantum computation.

In both classical and quantum computers, the use of reversible operations theoretically allows the energy required by the computer to be made arbitrarily small. In practice, energy is required to fight noise through restandardization and error correction. For example, memory of a measurement made during error detection must be erased before measuring again, which is an irreversible heat-generating process. Quantum systems currently under investigation are delicate and prone to error, and so the error-correcting infrastructure may well run quite hot – even hotter with gate constructions like that of Gottesman & Chuang (1999) that consume extra ancillary states. So the fact that quantum computing uses completely reversible logic does not imply that quantum computers will be energy efficient. In fact since quantum operators are currently applied by zapping a quantum system with lasers or other energy-guzzling instruments of torture to manipulate its effective system Hamiltonian, these smaller energy drains are unlikely to be a concern for some time.

The theory of quantum computing is racing far ahead of implementation, at times almost ludicrously so. But the developments in error correction and fault tolerance touched on in this paper, which have profound and immediate consequences for how gates can be organized and which of them make scalable basic building blocks, shows how important it is for theory to look a few steps ahead to guide implementation efforts.

References

Abrams, D. & Lloyd, S. (1998). Nonlinear quantum mechanics implies polynomial-time solution for NP-complete and #P problems, *Physical Review Letters* **81**(18): 3992–3995.

Aharonov, D. (1999). Quantum computation, *in* D. Stauffer (ed.), *Annual Reviews of Computational Physics VI*, World Scientific, pp. 259–346.

Albrecht, A. (1993). Following a "collapsing" wave function, *Physical Review D* **48**(8): 3768–3778.

Angelopoulos, A. et al. (1998). First direct observation of time reversal noninvariance in the neutral kaon system, *Physics Letters B* **444**: 43–51.

Barenco, A., Bennett, C. H., Cleve, R., DiVincenzo, D. P., Margolus, N. H., Shor, P. W., Sleator, T., Smolin, J. A. & Weinfurter, H. (1995). Elementary gates for quantum computation, *Physical Review A* **52**(5): 3457–3467.

Barnum, H., Caves, C. M., Finkelstein, J., Fuchs, C. A. & Schack, R. (2000). Quantum probability from decision theory?, *Proceedings of the Royal Society of London Series A* **456**(5): 1175–1182.

Benioff, P. (1982). Quantum mechanical models of Turing machines that dissipate no energy, *Physical Review Letters* **48**: 1581–1585.

Benioff, P. A. (1980). The computer as a physical system: a microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines, *Journal of Statistical Physics* **22**(5): 563–591.

Bennett, C., Bernstein, E., Brassard, G. & Vazirani, U. (1997). Strengths and weaknesses of quantum computation, *SIAM Journal on Computing* **26**: 1510–1523.

Bennett, C., Brassard, G., Crépeau, C., Jozsa, R., Peres, A. & Wootters, W. (1993). Teleporting an unknown quantum state via dual classical and Einstein-Podolsky-Rosen channels, *Physical Review Letters* **70**(13): 1895–1899.

Bennett, C., DiVincenzo, D., Smolin, J. & Wootters, W. (1996). Mixed-state entanglement and quantum error correction, **54**(5).

Bennett, C. H. (1973). Logical reversibility of computation, *IBM Journal of Research and Development* **17**(6): 525–532.

Bennett, C. H. (1982). The thermodynamics of computation– a review, *International Journal of Theoretical Physics* **21**: 905–940.

Bennett, C. H. (1988). Notes on the history of reversible computation, *IBM Journal of Research and Development* **32**(1): 16–23.

Bennett, C. H. (1989). Time/space trade-offs for reversible computation, *SIAM Journal on Computing* **18**(4): 766–776.

Bennett, C. H. (1993). Certainty from uncertainty, *Nature* **362**: 694–695.

Bennett, C. & Wiesner, S. (1992). Communication via one- and two-particle operators on Einstein-Podolsky-Rosen states, *Physical Review Letters* **69**(20).

Boykin, P. O., Mor, T., Pulver, M., Roychowdhury, V. & Vatan, F. (1999). On universal and fault-tolerant quantum computing, *Proceedings of the 40th Annual Symposium on Foundations of Computer Science*, pp. 486–494. Also quant-ph/9906054.

Brukner, C., Pan, J., Simon, C., Weihs, G. & Zeilinger, A. (2001). Probabilistic instantaneous quantum computation, *arXive e-print quant-ph/0109022*.

Chuang, I. L., Laflamme, R., Shor, P. W. & Zurek, W. H. (1995). Quantum computers, factoring and decoherence, *Science* **270**: 1635–1637. Also quant-ph/9503007.

Deutsch, D. (1985). Quantum theory, the Church-Turing principle and the universal quantum computer, *Proceedings of the Royal Society of London Series A* **400**: 97–117.

Deutsch, D. (1989). Quantum computational networks, *Proceedings of the Royal Society of London Series A* **A425**: 73–90.

Deutsch, D. (1999a). Quantum theory of probability and decisions, *Proceedings of the Royal Society of London Series A* **455**: 3129–3137. Also quant-ph/9906015.

Deutsch, D. (1999b). The structure of the multiverse. Preprint quant-ph/990600.

Deutsch, D. (2000). Information flow in entangled quantum systems, *Proceedings of the Royal Society of London Series A* **456**: 1759–1774. Also quant-ph/9906007.

Deutsch, D., Barenco, A. & Ekert, A. K. (1995). Universality in quantum computation, *Proceedings of the Royal Society of London Series A* **449**: 669–677. Also quant-ph/9505018.

Feynman, R. P. (1986). Quantum mechanical computers, *Foundations of Physics* **16**(6).

Frank, M. P. (1999). *Reversibility for Efficient Computing*, PhD thesis, MIT Artificial Intelligence Laboratory.

Fredkin, E. & Toffoli, T. (1982). Conservative logic, *International Journal of Theoretical Physics* **21**(3/4): 219–253.

Gershenfeld, N. (1996). Signal entropy and the thermodynamics of computation, *IBM Systems Journal* **35**(3&4): 577–586.

Gottesman, D. (1999). The Heisenberg representation of quantum computers, *in* S. P. Corney, R. Delbourgo & P. D. Jarvis (eds), *Proceedings of the XXII International Colloquium on Group Theoretical Methods in Physics, Group 22*, International Press, Cambridge, MA, pp. 32–43. Also quant-ph/9807006.

Gottesman, D. & Chuang, I. L. (1999). Demonstrating the viability of universal quantum computation using teleportation and single-qubit operations, *Nature* **402**: 390–392.

Greenberger, D., Horne, M., Shimony, A. & Zeilinger, A. (1990). Bell's theorem without inequalities, *American Journal of Physics* **58**: 1131–1143.

Grover, L. K. (1997). Quantum mechanics helps in searching for a needle in a haystack, *Physical Review Letters* **79**(2): 325–328.

Jozsa, R. (1997). Entanglement and quantum computation, *in* S. Huggett, L. Mason, K. P. Tod, S. T. Tsou & N. M. J. Woodhouse (eds), *Geometric Issues in the Foundations of Science*, Oxford University Press.

Jozsa, R. (1999). Quantum effects in algorithms, *Proceedings of first NASA Conference on Quantum Computation and Quantum Communication, Special Issue of Chaos and Solitons and Fractals*, pp. 1657–1664. Also quant-ph/9805086.

Knill, E., Laflamme, R. & Zurek, W. H. (1998). Resilient quantum computation, *Science* **279**: 342–345.

Landauer, R. (1961). Irreversibility and heat generation in the computing process, *IBM Journal of Research and Development* **5**(3): 183–191.

Lange, K., McKenzie, P. & Tapp, A. (1997). Reversible space equals deterministic space, *Proceedings of the 12th Annual IEEE Conference on Computational Complexity*, pp. 45–50.

Lecerf, Y. (1963). Machines de Turing réversibles, *Comptes Rendus des Séances de l'Académie des Sciences* **257**: 2597–2600. Translation available online (search keywords Michael Frank Lecerf Translation).

Leff, H. S. & Rex, A. F. (eds) (1990). *Maxwell's Demon: Entropy, Information, Computing*, Adam Hilger and Princeton University Press.

Levine, R. Y. & Sherman, A. T. (1990). A note on Bennett's time-space tradeoff for reversible computation, *SIAM Journal on Computing* **19**(4): 673–677.

Li, M., Tromp, J. & Vitanyi, P. (1998). Reversible simulation of irreversible computation, *Physica D* **120**: 168–176.

Li, M. & Vit'anyi, P. (1996). Reversibility and adiabatic computation: Trading time and space for energy, *Proceedings of the Royal Society of London, Series A* **452**: 769–789. Also quant-ph/9703022.

Lloyd, S. (1995). Almost any quantum logic gate is universal, *Physical Review Letters* **75**: 346–349.

Lloyd, S. (2000a). Quantum search without entanglement, *Physical Review A* **61**(010301).

Lloyd, S. (2000b). Ultimate physical limits to computation, *Nature* **406**: 1047–1054.

Margolus, N. & Levitin, L. (1998). The maximum speed of dynamical evolution, *Physica D* **120**(1-2): 188–195.

Nielsen, M. A., Knill, E. & Laflamme, R. (1998). Complete quantum teleportation using nuclear magnetic resonance, *Nature* **396**: 52–55.

Nielsen, M. & Chuang, I. (2001). *Quantum Computation and Quantum Information*, Cambridge University Press.

Preskill, J. (1998a). Quantum computing: pro and con, *Proceedings of the Royal Society of London Series A* **454**: 469–486. Also quant-ph/9705032.

Preskill, J. (1998b). Reliable quantum computers, *Proceedings of the Royal Society of London Series A* **454**(1969): 385–410.

Rieffel, E. G. & Polak, W. (2000). An introduction to quantum computing for non-physicists, *ACM Computing Surveys* **32**(3): 300–335.

Shor, P. W. (1995). Scheme for reducing decoherence in quantum memory, *Physical Review A* **52**(4): 2493–2496.

Shor, P. W. (1996). Fault-tolerant quantum computation, *Proceedings of the 37th Annual IEEE Symposium on Foundations of Computer Science*, pp. 56–65.

Steane, A. M. (1996). Error correcting codes in quantum theory, *Physical Review Letters* **77**(5): 793–797.

Szilard, L. (1929). Über die Entropieverminderung in einem thermodynamischen System bei Eingriffen intelligenter Wesen, *Zeitschrift für Physik* **53**: 840–856. Translated as 'On the Decrease of Entropy in a Thermodynamic System by the Intervention of Intelligent Beings' and republished in Leff & Rex (1990).

Toffoli, T. & Margolus, N. (1990). Invertible cellular automata: a review, *Physica D* **45**: 229–253.

Vandersypen, L. M. K., Steffen, M., Breyta, G., Yannoni, C. S., Sherwood, M. H. & Chuang, I. L. (2001). Experimental realization of Shor's quantum factoring algorithm using nuclear magnetic resonance, *Nature* **414**: 883–887.

Vieri, C. J. (1999). *Pendulum: A Reversible Computer Architecture*, PhD thesis, MIT Artificial Intelligence Laboratory.

Wootters, W. K. & Zurek, W. H. (1982). A single quantum cannot be cloned, *Nature* **299**: 802–803.

Yao, A. (1993). Quantum circuit complexity, *Proceedings of the 34th Annual Symposium on Foundations of Computer Science*, Institute of Electrical and Electronic Engineers Computer Society Press, Los Alamitos, CA, pp. 352–361.

Zhou, X., Leung, D. W. & Chuang, I. L. (2000). Methodology for quantum logic gate construction, *Physical Review A* **62**(052316).

References of the form quant-ph/0104033 are accessible at www.arXiv.org.