Building Quantum Computers

Plenary talk presented at the 2007 IEEE International Symposium on Information Theory, Toronto Canada

E. Knill

In theory, quantum computers can be used to efficiently factor numbers, quadratically speed up many search and optimization problems, and enable currently impossible physics simulations. At first, quantum states appeared to be too fragile for implementing large quantum computers. Fortunately, because of theoretical advances in quantum error correction and fault tolerance, there are now no fundamental obstacles to realizing quantum computers. However, building quantum computers is difficult. Current experiments can barely achieve adequate control of two quantum bits. Nevertheless, the gap between theoretical and practical quantum computing is closing. In what follows, I give a brief explanation of what quantum computers are, explain why we believe that in principle, arbitrarily large quantum computations can be accurately implemented, and survey the experimental state of the art and main implementation challenges.

A simple way to think of a quantum computer is as a traditional, *classical* computer with access to a quantum state machine. Thus quantum computing is an extension of classical computing with all the classical programming constructs available for problem solving and control of the quantum state machine. The quantum state machine is specified by its state space, initial state, transition operators and readout operators. It can be thought of as the result of applying the superposition principle to the 2^n *configurations* (bit strings) of n bit systems together with the ability to exploit interference, where n may vary during a computation. In particular, the state space consists of the unit vectors in a 2^n-dimensional Hilbert space with a distinguished orthonormal basis, whose elements are denoted by $|b\rangle$ with b bit strings of length n. The unit vectors can therefore be written as *superpositions* $|\psi\rangle = \sum_b \alpha_b |b\rangle$, where the complex numbers α_b are called the amplitudes of the superposition and $\sum_b |\alpha_b|^2 = 1$. For $n = 1$, the state space is that of a *quantum bit* (*qubit* for short). Just as bit strings of length n are the configurations of n bit systems, the superposition states $|\psi\rangle$ are considered to be states of n qubits. This is done by identifying the 2^n-dimensional Hilbert space with the tensor product of the n 2-dimensional Hilbert spaces associated with the qubits. The distinguished basis is obtained from the tensor products of the distinguished basis elements of the component qubits. Note that it is necessary to clearly distinguish between systems (such as qubits) and their states. This also makes it easier to understand the relationship between the formal definition of our state machines and their physical realizations.

The initial state of a quantum state machine has no qubits. To add qubits, we can make use of a transition operator that maps the state of n qubits $\sum_b \alpha_b |b\rangle$ to the state of $n+1$ qubits $\sum_b \alpha_b |b0\rangle$, where $b0$ is the length $n+1$ bit string obtained by appending 0. The representation of the states of a quantum state machine as the states of n qubits is important for defining unitary transition operators that may be applied to the states. One such operator is the Hadamard gate

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

acting on one qubit. H can be applied to the k'th of n qubits by tensoring with identity operators acting on the other $n-1$ qubits. Another is the Toffoli gate, which acts on three qubits by linear extension of the map on three bits that flips the third bit if the first two are 1. To define the linear extension, bit strings are identified with the corresponding distinguished basis elements. The Toffoli gate can be applied to three of n qubits by tensoring with identity operators acting on the remaining qubits. The Hadamard and Toffoli gates are sufficient for quantum computing. Nevertheless it is convenient to be able to apply any one-qubit unitary and the controlled-not gates. The controlled-not gate is the linear extension of the map that flips the second bit if the first one is 1. The Toffoli gate can be decomposed into a product of one-qubit unitary and controlled-not gates.

Information about the state of a quantum state machine is obtained by measurement. Suppose that the state of the machine is $\sum_b \alpha_b |b\rangle$. A full, destructive measurement returns the bit string b with probability $|\alpha_b|^2$ and resets the machine to its initial state. It is convenient, but not necessary, to be able to make nondestructive measurements of any one of the qubits. To learn how such measurements act, and for an introduction to quantum computing, see, for example, Nielsen and Chuang's textbook [10].

A phenomenon that is often mentioned as a source of the power of quantum computing is quantum parallelism, which involves the application of a classical reversible algorithm implemented by Toffoli gates "simultaneously" to all bit patterns in a superposition with exponentially many non-zero amplitudes. This is simply the generalization of the linear extension principle by which we defined the Toffoli gate. Transition operators such as the Hadamard gate must be used to prepare the state. Because the measurement cannot access amplitudes except by an exponentially complex analysis of the statistics of measurement outcomes, any use of such quantum parallelism must be followed by large scale interference of the state's amplitudes to extract the desired information. Interference refers to the effect by which one can reversibly increase amplitudes in some states in a way that is sensitive to relative phases. For example, the Hadamard gate applied to $|0\rangle$ yields the state $\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$, which when measured returns 0 or 1 with equal probability. Applying it again restores both the state $|0\rangle$ and determinism of the measurement outcome. If a process that changes the sign of the amplitude of $|1\rangle$ is applied before the second Hadamard gate, the final state is $|1\rangle$, demonstrating the sensitivity of the interference effect in the second Hadamard gate to the relative phases of the amplitudes. It is worth contrasting these effects to what is possible with probabilistic computing, where instead of superpositions involving amplitudes, we have *mixtures* involving probabilities of states. Gates correspond to Markov processes, which are reversible only if they are deterministic.

Building quantum computers requires physical systems with quantum state spaces that are capable of realizing qubit states and are sufficiently controllable. DiVincenzo [6] gives five requirements for the physical realization of quantum computers that correspond to the earlier specifications of a quantum state machine. The first is the availability of arbitrarily many independent quantum-information-carrying systems. The second requires that the quantum systems' state can be consistently intialized. Skipping the third requirement for now, the fourth asks for the ability to apply quantum operations sufficient for implementing arbitrary quantum computations efficiently. The fifth involves the ability to measure the systems so as to enable the required readout. These four requirements have been demonstrated individually in a number of physical systems.

The third and so far the most difficult requirement to demonstrate experimentally is that the states and operations are subject to sufficiently low noise. The continuous nature of the amplitudes and the sensitivity of interference effects to seemingly small changes in phases imply that quantum states and gates must be protected not only from bit flip errors, but from a continuous family of unwanted effects including changes in the phases of amplitudes. These unwanted effects are referred to as *decoherence*. Decoherence is associated with incomplete isolation from the environment and imperfect calibration of control fields required to implement gates. The microscopic nature of most suitable quantum systems and the need for strong interactions with the control and measurement apparatus makes it particularly difficult to reduce the effects of decoherence.

Like quantum gates, general quantum errors exhibit interference effects that preclude purely probabilistic models. Nevertheless, we commonly refer to gates having independent probabilities of error. This is justified if unwanted effects act independently in time and space and are unbiased. Although this is not generally the case, actions can be taken to increase independence and decrease bias. Alternatively, it is understood that the probability refers to the square of an amplitude in the operators expressing the effect of an error. Originally it was believed that in order to realize a quantum computation of size N, the probability of error per gate must be sufficiently smaller than $1/N^2$, where the square accounts for the possibility that errors add in amplitude rather than probability. However, as for classical computing with errors, it has been proven that under reasonable assumptions on the errors, if the probability of error per gate is smaller than some constant, then it is possible to efficiently quantum compute arbitrarily accurately. This result is known as the threshold theorem. See Sect. 10.6 of [10] for an overview of quantum fault tolerance and versions of this theorem. Since there are many ways to parameterize quantum error models and many physical constraints (such as spatial layout) to consider, the error threshold claimed by the theorem is best understood as defining a region of the relevant space of parameters and constraints where *scalable* quantum computing is possible in principle. Note that if the parameters are near the boundary of this region, the overhead required for implementing computations fault tolerantly becomes impractical.

Fault tolerant quantum computing involves using quantum error-detecting and -correcting codes to protect quantum information. To maintain and compute with the protected quantum information, we use carefully designed sequences of gates that ensure that any errors in the gates themselves do not disturb the protected information. All schemes for protecting quantum or classical information can be understood in terms of *subsystems*. Consider the trivial problem of protecting one qubit when we are given three physical qubits, where only the first two are subject to errors. The solution is to have the third qubit carry the protected information. The third qubit is a subsystem of the three-qubit physical system. Protected states are associated not with single states but with subspaces of states of the physical system, and the errors preserve these subspaces. Formally, a quantum subsystem of a physical system whose state space consists of unit states in the Hilbert space $\mathcal{H}$ is a tensor factor of a subspace of $\mathcal{H}$. The other factor is called the *cosubsystem*. Equivalently, finite quantum subsystems are characterized by subalgebras of the algebra of bounded operators on $\mathcal{H}$, where the subalgebras are ismorphic to matrix algebras. From a physical point of view, such subalgebras of operators consist of the (complex) *observables* of the subsystem and characterize the measurements that one can make of the states of the subsystem. The general scheme for protecting information is to determine a subsystem of the physical system that has the property that, provided the cosubsystem's state is suitably prepared, errors perturb only the cosubsystem's state with high probability. If the cosubsystem's state does not matter, then no action needs to be taken to maintain protection. Otherwise, it is necessary to periodically restore the cosubsystem to a state that ensures future protection. In the traditional view, this action is accomplished by error correction and re-encoding. From the subsystem view, the protected information never requires "correction"; it is sufficient to reset the cosubsystem after errors occurred. One can think of errors as increasing the entropy of the cosubsystems, and the protection procedure as a way of removing the entropy. Therefore, physical quantum computers generally require an entropy sink to protect information from errors.

The analysis of fault-tolerant quantum computing leads to strategies for eventually building large-scale quantum computers. Most suitable physical systems consist of localized quantum subsystems with at least two distinguishable states that can represent qubit states. These physically explicit qubits are normally subject to a significant amount of decoherence. The first task is to ensure sufficient control of the physical systems, including the ability to *couple* them, and to use whatever experimental techniques are available to reduce the effects of decoherence to the point where general-purpose error-correction techniques can be applied according to the threshold theorem. Eventually, fault-tolerant techniques are used to protect *logical* qubit subsystems that are nontrivially supported by many physical systems. Depending on the errors, it may be necessary to recursively construct qubit subsystems of lower-level logical qubits, a strategy known as *concatenation*. It helps to recognize that there are a number of common information processing tasks that are much easier to perform fault tolerantly than implementing unitary gates on logical qubits. These tasks include state preparation, measurement and quantum communication. In fact, the constraints on errors in physical operations used for these tasks are significantly weaker than on errors in unitary control. Thus, provided there is some means of establishing quantum communication channels between physical systems used to support logical qubits, one can initially focus on building very accurate quantum registers with only a small number (three or four) of qubits. One can rely on communication for computations requiring more qubits. Unlike classical communication, quantum communication and remote computation can be performed by what is known as quantum *teleportation*, which has the advantage of having no quantum latency. This implies that the speed of remote computation is not limited by slow quantum processes, only by the classical communication required for control. Although focusing on small but accurate quantum registers makes sense now, the ultimate goal is to ensure that good quantum gates are not much slower than classical circuit elements. This will require a tight integration of quantum and classical processing and fault tolerance.

Targeted experimental efforts to build quantum computers started with Shor's discovery of the quantum algorithm to factor large integers around 1994. Since then there have been many proposals to build quantum computers using a variety of physical systems. For a survey, see [1]. The clear current front runner for building small to medium size quantum computers is based on atomic qubits in ion traps. There are currently three other approaches that can claim to have demonstrated coherent two-qubit control: Liquid state nuclear magnetic resonance (NMR) quantum computing, postselected photonic qubits, and superconducting qubits. Of these approaches, the first two have little hope of constructing quantum registers with more than about ten qubits, because of inherent exponential inefficiencies that require a significant change or addition to the underlying technology.

To be able to usefully solve problems currently infeasible on classical computers with known quantum algorithms requires thousands of qubits and billions of gates. Although up to eight qubits have been nontrivially manipulated with atomic qubits in ion traps, at this point no one has clearly demonstrated a computationally useful two-qubit register. It is expected that this will be achieved shortly in ion traps.

In ion-trap quantum computing, the physical qubits are represented by two energy levels of ions that are electromagnetically trapped. The ions can be manipulated by means of laser pulses. The combination of the trapping potential and Coulomb repulsion leads to common vibrational modes that can be exploited for applying nontrivial two-qubit gates. This approach to quantum computing can be scaled by having multiple traps with the ability to move ions between them as proposed by Wineland and coauthors [12]. All but the requirement for sufficiently low noise have been individually demonstrated. There are three main challenges for experimental ion-trap quantum computing. The first is to realize gates with sufficiently low error probabilities. Error probabilities of about 0.5% have been demonstrated for two-qubit gates [2]. The current guidelines for demonstration of the low-noise requirement are to have less than 0.01% probability of error per unitary gate. State preparation and measurement can have probabilities of error of 1%, which has been demonstrated in ion traps. The second challenge is to show that all the requirements can be met in one device. This is a problem of technology integration and is typically much harder than demonstrating each requirement independently. The third challenge is to have an efficient way of quantum communicating between ion-trap quantum registers, preferably by optical interconnects. The first steps in this direction have been taken by Moehring and coauthors [8].

Superconducting qubits are based on the collective dissipation-less behavior of electrons in superconducting circuits. There are a number of different ways to design such circuits to exhibit the desired two-level subsystems needed to represent qubits. For reviews of the relevant physics, see [4], [5]. It was not clear whether the collective effects were experimentally accessible until some coherent control and measurement of qubits in superconducting circuits was demonstrated by Nakamura and coworkers [9]. Unexpectedly, experiental quantum computing with superconducting qubits is progressing rapidly and has overtaken other seemingly more promising approaches. A possible advantage of superconducting qubits is that it is possible to have gates that are much faster than is practical with atomic qubits. Because noise also acts on shorter time scales, this is also a challenge, requiring high-quality control involving very fast electronics. At this time, slow gates are an advantage as the electronics required for control is off-the-shelf. The path toward large scale quantum computing with superconducting qubits is not yet as well defined as for atomic qubits in ion traps, so the requirements have not been demonstrated as clearly. Because current realizations of superconducting qubits require temperatures well below 1 K, the challenge of integrating technology seems more severe at the moment. Communication with devices in separate refrigeration units is also difficult and no means for doing so has been demonstrated so far.

There are many other approaches to building quantum computers that are being investigated experimentally. Promising ones include atomic qubits of trapped atoms in optical lattices [3] and various quantum-dot-based schemes [7], both of which have two-qubit gate demonstrations in progress. There are also esoteric approaches, such as topological quantum computing based on anyonic excitations, which is claimed to be intrinsically robust against noise. Whether and where these excitations can be found in experimentally accessible condensed matter phases is a subject of theoretical controversy and experimental investigation [11].

Since the challenge of building quantum computers has no analogue in the history of computing, this is a great time to be doing research in quantum technologies. There are many theoretical and experimental problems to be solved and challenges to be met, and although difficult, they are likely surmountable. The associated improvements in quantum control have wide applicability beyond quantum computing proper. Assuming no fundamental physics surprises, which would of course be welcome, I expect the use of quantum mechanics in practical technology and computation to become pervasive.

Acknowledgments

This is a contribution of the National Institute of Standards and Technology, an agency of the U.S. government, and is not subject to U.S. copyright.

References

[1] Special issue on implementations of quantum computers. *Fort. Phys.*, vol. 48, no. 9–11, 2000.

[2] J. Benhelm, G. Kirchmair, C.F. Roos, and R. Blatt, “Towards fault-tolerant quantum computing with trapped ions,” *Nature Phys.*, vol. 4, pp. 463–466, 2008.

[3] I. Bloch, “Quantum coherence and entanglement with ultracold atoms in optical lattices,” *Nature*, vol. 453, pp. 1016–1022, 2008.

[4] J. Clarke and F. Wilhelm, “Superconducting quantum bits,” *Nature*, vol. 453, pp. 1031–1042, 2008.

[5] M.H. Devoret, A. Wallraff, and J.M. Martinis, Superconducting qubits: A short review. quant-ph / 0411174, 2004.

[6] D.P. DiVincenzo, “The physical implementation of quantum computation,” *Fort. Phys.*, vol. 48, pp. 771–783, 2000.

[7] R. Hanson and D.D. Awschalom, “Coherent manipulation of single spins in semiconductors,” *Nature*, vol. 453, pp. 1043–1049, 2008.

[8] D.L. Moehring, P. Maunz, S. Olmschenk, K.C. Younge, D.N. Matsukevich, L.-M. Duan, and C. Monroe, “Entanglement of single-atom quantum bits at a distance,” *Nature*, vol. 449, pp. 68–71, 2007.

[9] Y. Nakamura, Y.A. Pashkin, and J.S. Tsai, “Coherent control of macroscopic quantum states in a single-cooper-pair box,” *Nature*, vol. 398, pp. 786–788, 1999.

[10] M.A. Nielsen and I.L. Chuang, *Quantum Computation and Quantum Information*. Cambridge University Press, Cambridge, UK, 2001.

[11] S. Das Sarma, M. Freedman, and C. Nayak, “Topological quantum computation,” *Phys. Today*, vol. 59, pp. 32–38, 2006.

[12] D.J. Wineland, C. Monroe, W.M. Itano, D. Leibfried, B.E. King, and D.M. Meekhof, “Experimental issues in coherent quantum-state manipulation of trapped atomic ions,” *J. Res. Nat. Inst. St. Tech.*, vol. 103, pp. 259–328, 1998.