Fault-tolerant quantum computing

John Preskill
APS March Meeting
Denver, 4 March 2014
Quantum Computation

Feynman ‘81  Deutsch ‘85  Shor ‘94
A computer that operates on quantum states can perform tasks that are beyond the capability of any conceivable classical computer.

Feynman ‘81  Deutsch ‘85  Shor ‘94
Finding Prime Factors

\[1807082088687 \times 4048059516561 \times 6440590556627 \times 8102516769401 \times 3491701270214 \times 5005666254024 \times 4048387341127 \times 5908123033717 \times 8188796656318 \times 2013214880557 \]
Finding Prime Factors

The boundary between “hard” and “easy” seems to be different in a quantum world than in a classical world.
Ron Rivest, Adi Shamir, Len Adleman
What’s in here?
Quantum algorithms

Exploring complexity: We should be able to check (someday) that quantum physics exploits extravagant resources by verifying superpolynomial speedups for (NP) problems where solution can be checked classically, like factoring. (However, there is no proof that factoring is hard classically.)

Not NP-hard (in the worst case): Superpolynomial quantum speedups seem to be possible only for problems with special structure, not for NP-complete problems like 3-SAT. Quantum physics speeds up unstructured search quadratically, not exponentially.

Beyond NP: Speedups for problems outside NP are also common and important. Indeed the “natural” application for a quantum computer is simulating time evolution of quantum systems, e.g. collisions in molecular chemistry or quantum field theory.

In such cases the findings of a quantum computer might not be easy to check with a classical computer; instead, one quantum computer must be checked by another (or by doing an experiment, which is sort of the same thing).
Toward quantum supremacy

We are now at a unique stage in human history, where we can envision, and have the experimental tools to achieve, building and controlling larger and larger systems which behave in an intrinsically quantum mechanical way.

The quantum computing adventure will enter the new, more mature phase of “quantum supremacy” once we can prepare and control complex quantum systems that behave in ways that cannot be predicted using digital computers (systems that “surpass understanding” and surprise us).

Who knows what we’ll find?
Why quantum computing is hard

We want qubits to interact strongly with one another.

We don’t want qubits to interact with the environment.

Until we measure them.
Decoherence

\[ \frac{1}{\sqrt{2}} ( \text{Environment} ) \]
Decoherence explains why quantum phenomena, though observable in the microscopic systems studied in the physics lab, are not manifest in the macroscopic physical systems that we encounter in our ordinary experience.
How can we protect a quantum computer from decoherence and other sources of error?
Quantum Error Correction

Shor ‘95

Steane ‘95
Quantum information can be protected, and processed fault-tolerantly.

Shor ‘95

Steane ‘95
Quantum error correction

Protect not just against bit flips, but also against the environment “watching the computer,” so that computational paths can interfere.

If a quantum computation works, and you ask the quantum computer later what it did, it should answer: “I forget..”

The computation is encrypted, i.e. hidden from the environment. (Not the answer, which is classical, but the path followed by the computer to reach the answer.)

And even a properly “encrypted” computation may fail, unless the gates are sufficiently accurate.

Irony: Macroscopic systems are usually highly vulnerable to decoherence, but we can protect information better by encoding it nonlocally, in a “macroscopic” memory.
Alexei Kitaev
9 April 1997 … An exciting day!

A. Kitaev

Classical Fault Tolerance
- Not needed! Why?

Magnetic disk:
\[ H = - J \sum \hat{S}_i \cdot \hat{S}_{i+1} \]  
(spin alignment)

Rep. code has no quantum analog

Closest thing is “toric code”

Forms:
- qudits on edges of lattice
- Stabilizer generators:
  \[ A_r = \prod_j \hat{S}_j \]  
  \[ B_z = \prod_j \hat{S}_j \]  

All mutually commuting
Topology

Quantum Computer

Noise!
Aharonov-Bohm Phase $\Phi$

$\exp(i \epsilon \Phi)$
Nonabelian anyons

Quantum information can be stored in the collective state of exotic particles in two spatial dimensions (“anyons”).

The information can be processed by exchanging the positions of the anyons (even though the anyons never come close to one another).
Nonabelian anyons

Quantum information can be stored in the collective state of exotic particles in two spatial dimensions (“anyons”).

The information can be processed by exchanging the positions of the anyons (even though the anyons never come close to one another).
$Z_2$ Aharonov-Bohm Phase

\[ \text{Diagram} \]
$Z_2$ Aharonov-Bohm Phase

\[ -1 \]
Topological Degeneracy

A two-dimensional system (with a mass gap) that supports quasiparticle excitations with nontrivial Aharonov-Bohm interactions has a ground state degeneracy that depends on the topology of the surface.

Example: two defects (green and red) with a $\mathbb{Z}_2$ Aharonov Bohm phase. **Green** defects can be singly produced or annihilated at a green boundary, **red** defects can be singly produced or annihilated at a red boundary.

Two operators ($R$ and $G$) both preserve the ground state, and obey a nontrivial commutation relation: $R^{-1}G^{-1}RG = -1$.

This algebra has no one-dimensional representations, hence the ground state is (two-fold) degenerate.
Topological Degeneracy

A two-dimensional system (with a mass gap) that supports quasiparticle excitations with nontrivial Aharonov-Bohm interactions has a ground state degeneracy that depends on the topology of the surface.

Example: two defects (green and red) with a $Z_2$ Aharonov Bohm phase. Green defects can be singly produced or annihilated at a green boundary, red defects can be singly produced or annihilated at a red boundary.

Two operators ($R$ and $G$) both preserve the ground state, and obey a nontrivial commutation relation: $R^{-1}G^{-1}RG = -1$. This algebra has no one-dimensional representations, hence the ground state is (two-fold) degenerate.
Surface codes \textbf{(Kitaev '96)}

Qubits are arranged at the links of a square lattice on a two-dimensional surface, and the check operators are four-qubit operators that can be measured \textit{locally}:

\[
\prod X = 1, \quad \prod Z = 1, \\
X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}
\]

These operators are mutually commuting (and so can be simultaneously diagonalized). The code space is analogous to the ground state of a $Z_2$ gauge theory --- the site operators detect electric charges and the plaquette operators detect $Z_2$ magnetic flux.
Robustness of stored quantum information can be attributed to long-range Aharonov-Bohm interaction between \textit{site defect} (electric charge) and \textit{plaquette defect} (magnetic flux).

There is an associated topological ground state degeneracy.

A coherent superposition of ground states can be disturbed only if a (site or plaquette) defect travels across the sample, i.e, on a homologically nontrivial path.
On the torus: The ground state code $\mathcal{H}_{\text{code}}$ (no defects) is preserved by $\prod_{C} \mathbb{Z}$ if $C$ is a cycle (has no boundary).

Homologically trivial; $\prod_{C} \mathbb{Z}$ is a product of check operators. It acts trivially on the code space.

Homologically nontrivial; $\prod_{C} \mathbb{Z}$ is not a product of check operators. It is a logical operation acting on the code subspace.
Code on a planar surface with boundary

3-qubit plaquette check operators at rough edges and 3-qubit site check operators at smooth edges.

Defects can fall off the end of the world.
Code on a planar surface with boundary

3-qubit plaquette check operators at rough edges and 3-qubit site check operators at smooth edges.

Defects can fall off the end of the world.
Code on a planar surface with boundary

3-qubit plaquette check operators at rough edges and 3-qubit site check operators at smooth edges.

Defects can fall off the end of the world.
Code on a planar surface with boundary

3-qubit plaquette check operators at rough edges and 3-qubit site check operators at smooth edges.

Defects can fall off the end of the world.
Code on a planar surface with boundary

3-qubit plaquette check operators at rough edges and 3-qubit site check operators at smooth edges.

Defects can fall off the end of the world.
Code on a planar surface with boundary

3-qubit plaquette check operators at rough edges and 3-qubit site check operators at smooth edges.

Defects can fall off the end of the world.
Code on a planar surface with boundary

3-qubit plaquette check operators at rough edges and 3-qubit site check operators at smooth edges.

Defects can fall off the end of the world.
Code on a planar surface with boundary

3-qubit plaquette check operators at rough edges and 3-qubit site check operators at smooth edges.

Defects can fall off the end of the world.
Code on a planar surface with boundary

3-qubit plaquette check operators at rough edges and 3-qubit site check operators at smooth edges.

Defects can fall off the end of the world.
Code on a planar surface with boundary

Logical operations

Encoded operations are associated with homology cycles \textit{relative to the boundary} --- i.e., paths that run from one edge to another of the same type.

Defects can fall off the end of the world.
Toric Code Recovery

Measuring the syndrome detects the endpoints of a chain of errors.
Measuring the syndrome detects the endpoints of a chain of errors.

The syndrome is highly ambiguous; many different error chains can have the same boundary.

But as long as our interpretation of the syndrome is *homologically* correct, recovery will be successful.
Measuring the syndrome detects the endpoints of a chain of errors.

The syndrome is highly ambiguous; many different error chains can have the same boundary.

But as long as our interpretation of the syndrome is *homologically* correct, recovery will be successful.
Measuring the syndrome detects the endpoints of a chain of errors.

The syndrome is highly ambiguous; many different error chains can have the same boundary.

But as long as our interpretation of the syndrome is *homologically* correct, recovery will be successful.
Toric Code Recovery

ERROR!
Z-type errors create / annihilate pairs of electric charges, or move an electric charge to a neighboring site. X-type errors create / annihilate pairs of magnetic charges or move magnetic charges. If the error rate is small, the error chain segments are typically short, and the particle positions are strongly correlated. Once the particle positions are known, it is “easy” to guess how to bring particles together and annihilate them without a logical error. Measurements of particle positions are sometimes wrong, but we can repeat measurements to make our guess reliable.

There is an “accuracy threshold”

... if we assume accurate and instantaneous (poly-time) classical processing. The probability of a logical error decays exponentially with system size.

Dennis, Kitaev, Landahl, Preskill (2002).
Syndrome measurement circuits

To estimate the critical error rate for two-qubit gates, we must analyze the syndrome measurement circuits. A single error in the circuit can produce two in the data, which complicates the analysis...

\[ \epsilon_0 \approx 0.75 \times 10^{-2} \]

(assuming local gates, parallel operation, fast measurements, fast classical processing...and uncorrelated errors)
To encode many qubits in a planar system, consider a surface code with many “punctures”.

For example, it may be possible for an electric defect to be “hidden” inside a puncture (it can be detected by measuring the Aharonov-Bohm phase acquired by a magnetic charge carried around the puncture).

Thus, logical (“string”) operations are realized by carrying a magnetic charge (red) around an electric hole, or by moving an electric charge (green) from one hole to another. Quantum information is well protected if punctures are large and far apart.

In a single planar system we may also use two different dual encodings: Some punctures carry electric charges while others carry magnetic charges.
Topological quantum gates

We can perform a CNOT gate by exploiting the $Z_2$ A-B interaction between electric and magnetic punctures. (CNOT is a phase if the control is in the Z basis and target is in the X basis.)

The punctures are braided by a sequence of local code deformations.

We can “teleport” from the electric encoding to the magnetic encoding, braid the punctures, and then teleport back.

Raussendorf 2005
Surface codes: Towards practical large-scale quantum computation

Austin G. Fowler
Centre for Quantum Computation and Communication Technology, School of Physics, The University of Melbourne, Victoria 3010, Australia

Matteo Mariantoni
Department of Physics, University of California, Santa Barbara, CA 93106-9530, USA and California Nanosystems Institute, University of California, Santa Barbara, CA 93106-9530, USA

John M. Martinis and Andrew N. Cleland
California Nanosystems Institute, University of California, Santa Barbara, CA 93106-9530, USA
(Dated: October 26, 2012)

This article provides an introduction to surface code quantum computing. We first estimate the size and speed of a surface code quantum computer. We then introduce the concept of the stabilizer, using two qubits, and extend this concept to stabilizers acting on a two-dimensional array of physical qubits, on which we implement the surface code. We next describe how logical qubits are formed in the surface code array and give numerical estimates of their fault-tolerance. We outline how logical qubits are physically moved on the array, how qubit braid transformations are constructed, and how a braid between two logical qubits is equivalent to a controlled-NOT. We then describe the single-qubit Hadamard, $\hat{S}$ and $\hat{T}$ operators, completing the set of required gates for a universal quantum computer. We conclude by briefly discussing physical implementations of the surface code. We include a number of appendices in which we provide supplementary information to the main text.
To factor 2048 bits: ~ 10K logical qubits and ~ 1B physical (superconducting) qubits with $p \sim 10^{-3}$ per gate; runs in ~ 1 day.

Progress:

Better quantum compilers to reduce logical gate count.

Improved methods for decoding (i.e. error correcting) logical blocks.

Improved methods for completing universal gate set (e.g. doing a Toffoli gate). State distillation dominates overhead.
Session Index

Session Q32: Invited Session: Quantum Computing Architectures

Sponsoring Units: GQI
Chair: Andrew Landahl, Sandia National Laboratories
Room: 708 712

Wednesday, March 5, 2014
2:30PM - 3:06PM
Q32.00001: Quantum Compiling for Topological Quantum Computing
Invited Speaker: Krysta Svore
Preview Abstract

Wednesday, March 5, 2014
3:06PM - 3:42PM
Q32.00002: Error correction for adiabatic quantum computing
Invited Speaker: Kevin Young
Preview Abstract

Wednesday, March 5, 2014
3:42PM - 4:18PM
Q32.00003: Architectures for measurement-based quantum computation
Invited Speaker: Robert Raussendorf
Preview Abstract

Wednesday, March 5, 2014
4:18PM - 4:54PM
Q32.00004: Synthesizing Logic in Fault-Tolerant Quantum Computers
Invited Speaker: Cody Jones
Preview Abstract

Wednesday, March 5, 2014
4:54PM - 5:30PM
Q32.00005: TBD
Invited Speaker: Olivier Landon-Cardinal
Preview Abstract
Session L35: Focus Session: Quantum Computing Architectures and Algorithms: Quantum Error Correction

Sponsoring Units: QCI
Room: 702

Wednesday, March 5, 2014
8:00AM - 8:12AM
L35.00001: Fibre bundle framework for quantum fault tolerance
Lucy Liuxuan Zhang, Daniel Gottesman
Preview Abstract

Wednesday, March 5, 2014
8:12AM - 8:24AM
L35.00002: Magic-state encoder and magic teleportation: Efficient fault-tolerant non-Clifford gates with concatenated quantum codes
Hayato Goto, Satoshi Nakamura, Mamiko Kujiroka, Kouichi Ichimura
Preview Abstract

Wednesday, March 5, 2014
8:24AM - 8:36AM
L35.00003: Quantum computing with low overhead
Guillaume Duclos-Cianci, David Poulin
Preview Abstract

Wednesday, March 5, 2014
8:36AM - 9:12AM
L35.00004: The overhead of fault-tolerant quantum computing
Invited Speaker: Daniel Gottesman
Preview Abstract

Wednesday, March 5, 2014
9:12AM - 9:24AM
L35.00005: Extremal Optimization for estimation of the error threshold in topological subsystem codes
Jorge E. Millan-Otoya, Stefan Boettcher
Preview Abstract
A quantum computer can solve hard problems - such as prime factoring\cite{1}, database searching\cite{2}, and quantum simulation\cite{3} - at the cost of needing to protect fragile quantum states from error. Quantum error correction\cite{4} provides this protection, by distributing a logical state among many physical qubits via quantum entanglement. Superconductivity is an appealing platform, as it allows for constructing large quantum circuits, and is compatible with microfabrication. For superconducting qubits the surface code\cite{5} is a natural choice for error correction, as it uses only nearest-neighbour coupling and rapidly-cycled entangling gates. The gate fidelity requirements are modest: The per-step fidelity threshold is only about 99%. Here, we demonstrate a universal set of logic gates in a superconducting multi-qubit processor, achieving an average single-qubit gate fidelity of 99.92% and a two-qubit gate fidelity up to 99.4%. This places Josephson quantum computing at the fault-tolerant threshold for surface code error correction. Our quantum processor is a first step towards the surface code, using five qubits arranged in a linear array with nearest-neighbour coupling. As a further demonstration, we construct a five-qubit Greenberger-Horne-Zeilinger (GHZ) state\cite{6} using the complete circuit and full set of gates. The results demonstrate that Josephson quantum computing is a high-fidelity technology, with a clear path to scaling up to large-scale, fault-tolerant quantum circuits.

We characterise our gate fidelities using Clifford-based randomised benchmarking\cite{7,8,9}. The Clifford group is a set of rotations that evenly samples the Hilbert space, thus averaging across errors. For the single-qubit case the Cliffords are comprised of $\pi$, $\pi/2$ and $2\pi/3$ rotations, see Supplementary Information. In randomised benchmarking, a logic gate is characterised by measuring its performance when interleaved with

Two-qubit controlled Z-gate in 40 ns with F = 99.4%
Single-qubits gates with (average) F = 99.92%
Two Physical Systems

What is the difference between:

A: Human

Noisy hardware.
Information processing prevents information *loss*.

B: Chip

Reliable hardware.
Error correction less essential to its operation.
Protected superconducting qubit

Physically robust encodings have been proposed using superconducting circuits containing Josephson junctions, for example the “0-Pi qubit”. The circuit’s energy $E(\theta)$, as a function of the superconducting phase difference $\theta$ between its leads, is a periodic function with period $\pi$ to an excellent approximation.

“0-Pi qubit”:

$$E \approx f(2\theta) + O\left(\exp\left(-c(\text{size})\right)\right)$$

Two states localized near $\theta=0$ and $\theta=\pi$ are the basis states of a protected qubit. The barrier is high enough to suppress bit flips, and the stable degeneracy suppresses phase errors. Protection arises because the encoding of quantum information is highly nonlocal, and splitting of degeneracy scales exponentially with size of the device.
For reliable quantum computing, we need not just very stable qubits, but also the ability to apply very accurate nontrivial quantum gates to the qubits.

Accurate (Clifford group) phase gates can be applied to 0-Pi qubits by turning on and off the coupling between a qubit (or pair of qubits) and a harmonic oscillator (an LC circuit whose inductance is large in natural units). In principle the gate error becomes exponentially small as the inductance grows.

The reliability of the gate arises from a continuous-variable quantum error-correcting code underlying its operation, in which a qubit is embedded in the infinite-dimensional Hilbert space of a harmonic oscillator. Coupling the 0-Pi qubit to the oscillator sends the oscillator on a state-dependent phase space excursion during which it acquires a geometric phase that is protected by the code.
Protected phase gate

\[ \exp \left( i \frac{\pi}{4} Z \right) \]

Switch is really a tunable Josephson junction:

\[ H = \frac{Q^2}{2C} + \frac{\varphi^2}{2L} - J(t) \cos(\varphi - \theta) \]

Under suitable adiabaticity conditions, closing the switch transforms a broad oscillator state (e.g. the ground state) into a grid state (approximate codeword).

Peaks are at even or odd multiples of \( \pi \) depending on whether \( \theta \) is 0 or \( \pi \), i.e. on whether qubit is 0 or 1. Inner width squared is \( (JC)^{-1/2} \) and outer width is \( (L/C)^{1/2} \)

\[ \omega_j^{-1} = \sqrt{C/J} \ll \text{switching time} \ll \omega^{-1} = \sqrt{LC} \gg 1 \]
The calculable contribution to error due to diabatic effects and $Q$-space spreading is approximately:

$P_{\text{error}}^{(l+)}(\varepsilon) = \left( \frac{L}{C} \right)^{1/2} = 80$

$\left( \frac{JC}{L} \right)^{1/2} = 8$

$\tau_j / C = 80$
Large inductance

The intrinsic error scales like $\exp\left(-\frac{1}{4}\sqrt{L/C}\right)$.

Is $\sqrt{L/C} \approx 80$ reasonable?

Manucharyan et al. 2009, Masluk et al. 2012, Bell et al. 2012 achieved ~ 20 with a chains of Josephson junctions. The inductance scales linearly with the length of the chain, but there are potential obstacles to building very long chains. Another possible approach is to exploit the large (kinetic) inductance in amorphous superconductors.

What about universal quantum computation and measurement?

-- If we can prepare and measure in the basis $|0\rangle \pm |1\rangle$, a noisy $\pi/4$ single-qubit phase gate ($F > .93$), augmented by state distillation, suffices for fault-tolerant universality (Bravyi & Kitaev 2005).

-- It is also okay if measurements are noisier than gates, as we can protect measurements using repetition (or coding)

-- So if we can really do a two-qubit phase gate with high fidelity, that’s worth a lot!
Scalability of quantum computing

“Scalability” means that we can solve problems using resources that scale reasonably with the size of the input to the problem. The accuracy threshold theorem for quantum computation establishes that scalability is achievable provided that:

(1) the currently accepted principles of quantum physics hold,

and

(2) the noise afflicting a quantum computer is neither too strong nor too strongly correlated.

For scalability to fail as a matter of principle then, either:

(1) quantum mechanics fails for complex highly entangled systems (e.g., ‘t Hooft), or

(2) the Hamiltonian or the quantum state of the world imposes noise correlations that overwhelm fault-tolerant quantum protocols (e.g., Alicki, Kalai).

Skepticism is natural and useful. Skeptics should be pressed for a conception of Nature in which classical computing is feasible yet quantum computing is forbidden.