A brief overview of quantum computing

or,

Can we compute faster in a multiverse?

Tom Carter

https://csustan.csustan.edu/~ tom/Lecture-Notes/Quantum-Computing/qc-article.pdf

July 16, 2005

Our general topics: ←

- Hilbert space and quantum mechanics
- Tensor products
- Quantum bits (qubits)
- Entangled quantum states
- Quantum computing
- Simple quantum gates
- Tractability of computation
- Factoring
- Notes on factoring
- Quantum algorithms for satisfiability
- Possibilities for physical implementation
- Decoherence and error correction
- Prospects
- References
- On-line references

The quotes ↞

- ◎ Twelve men
- ◎ Magic
- ◎ Shocking
- ◎ Finis

To top ←

Twelve men

↞

”There was a time when the newspapers said that only 12 men understood the theory of relativity. I do not believe there ever was such a time. There might have been a time when only 1 man did, because he was the only guy who caught on, before he wrote his paper. But after people read the paper a lot of people understood the theory of relativity in some way or other, certainly more than 12. On the other hand, I think I can safely say that nobody understands quantum mechanics”

-Richard Feynman

Hilbert space and quantum mechanics $\leftarrow$

- A Hilbert space H is a complete normed vector space over $\mathbb{C}$:

1. H is a vector space over $\mathbb{C}$

2. There is an inner product
   $\langle \cdot | \cdot \rangle$: $H \times H \to \mathbb{C}$
   which is conjugate linear:
   $\langle v|w\rangle = \overline{\langle w|v\rangle}$
   $\langle \alpha v|w\rangle = \alpha\langle v|w\rangle$ for $\alpha \in \mathbb{C}$
   $\langle v+w|z\rangle = \langle v|z\rangle + \langle w|z\rangle$
   $\langle v|v\rangle \geq 0$ and $\langle v|v\rangle = 0$ iff $v = 0$

3. From the inner product, as usual, we define the norm of a vector:
   $\|v\|^2 = \langle v|v\rangle$

4. H is complete with respect to the norm.

- We will typically use the bra/ket notation: $|v\rangle$ is a vector in H, and $\langle v|$ is the covector which is the conjugate transpose of v.

- This notation also allows us to represent the outer product of a vector and covector as $|v\rangle\langle w|$, which, for example, acts on a vector $|z\rangle$ as $|v\rangle\langle w|z\rangle$. For example, if $\{v_1,v_2\}$ is an orthonormal basis for a two-dimensional Hilbert space, $|v_1\rangle\langle v_2|$ is the transformation that maps $|v_2\rangle$ to $|v_1\rangle$ and $|v_1\rangle$ to $(0,0)^T$ since

  $$|v_1\rangle\langle v_2||v_2\rangle = |v_1\rangle\langle v_2|v_2\rangle = |v_1\rangle$$
  $$|v_1\rangle\langle v_2||v_1\rangle = |v_1\rangle\langle v_2|v_1\rangle = 0|v_1\rangle = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

  Equivalently, $|v_1\rangle\langle v_2|$ can be written in matrix form where $|v_1\rangle = (1,0)^T$, $\langle v_1| = (1,0)$, $|v_2\rangle = (0,1)^T$, and $\langle v_2| = (0,1)$. Then

  $$|v_1\rangle\langle v_2| = \begin{pmatrix} 1 \\ 0 \end{pmatrix}(0,1) = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}.$$

- A unitary operator $U : H \to H$ is a linear mapping whose conjugate transpose is its inverse: $U^\dagger = U^{-1}$

- Unitary operators are norm preserving: $\|Uv\|^2 = \langle v|U^\dagger U|v\rangle = \langle v|v\rangle = \|v\|^2$

- We will think of a quantum state as a (normalized) vector $|v\rangle \in H$. For math folks, we are in effect working in Complex projective space, normalizing to 1 so that the probabilities make sense.

- The dynamical evolution of a quantum system is expressed as a unitary operator acting on the quantum state.

- Eigenvalues of a unitary matrix are of the form $e^{i\omega}$ where ω is a real-valued angle. A unitary operator is in effect a rotation.

- Just for reference, a typical expression of Schrödinger's equation looks like

$$\left[-\frac{\hbar^2}{2m_e}\bigtriangledown^2 + V(x,y,z)\right]\Psi = i\hbar\frac{\partial}{\partial t}\Psi$$

  with general solution

$$\Psi(x,y,z,t) = \sum_{n=0}^{\infty} c_n \Psi_n(x,y,z) \exp\left(\frac{-iE_n t}{\hbar}\right)$$

  where $\Psi_n(x,y,z)$ is an eigenfunction solution of the time independent Schrödinger equation with E_n the corresponding eigenvalue. The inner product, giving a time dependent probability, looks like

$$P(t) = \int \overline{\Psi}\Psi dv.$$

- Another way to think of this is that we have to find the Hamiltonian $\mathcal{H}$ which generates evolution according to:

$$i\hbar\frac{\partial}{\partial t}|\Psi(t)\rangle = \mathcal{H}|\Psi(t)\rangle.$$

In our context, we will have to solve for $\mathcal{H}$ given a desired U:

$$|\Psi_f\rangle = \exp\left(-\frac{i}{\hbar}\int \mathcal{H} dt\right)|\Psi_0\rangle = U|\Psi_0\rangle$$

A solution for $\mathcal{H}$ always exists, as long as the linear operator U is unitary.

- A measurement consists of applying an operator O to a quantum state v. To correspond to a classical observable, O must be *Hermitian*, $O^\dagger = O$, so that all its eigenvalues are real. If one of its eigenvalues λ is associated with a single eigenvector u_λ, then we observe the value λ with probability $|\langle v|u_\lambda\rangle|^2$ (i.e., the square of the length of the projection along u_λ).

- In general, if there is more than one eigenvector u_λ associated with the eigenvalue λ, we let P_λ be the projection operator onto the subspace spanned by the eigenvectors, and the probability of observing λ when the system is in state v is $\|P_\lambda v\|^2$.

- Most projection operators do not commute with each other, and are not invertible. Therefore, we can expect that the order in which we do measurements will matter, and that doing a measurement will irreversibly change the state of the quantum system.

Tensor products

←

- We can form tensor products of a wide variety of objects. For example:

  1. The tensor product of an n dimensional vector u and an m dimensional vector v is an nm dimensional vector $u \otimes v$.

  2. If A and B are operators on n and m dimensional vectors, respectively, then $A \otimes B$ is an operator on nm dimensional vectors.

  3. if H_1 and H_2 are Hilbert spaces, then $H_1 \otimes H_2$ is also a Hilbert space. If H_1 and H_2 are finite dimensional with bases $\{u_1, u_2, \ldots u_n\}$ and $\{v_1, v_2, \ldots v_m\}$ respectively, then $H_1 \otimes H_2$ has dimension nm with basis $\{u_i \otimes v_j | 1 \leq i \leq n, 1 \leq j \leq m\}$.

- Tensor products obey a number of nice rules. For matrices A, B, C, D, U, vectors u, v, w, and scalars a, b, c, d the following hold:

  $$\begin{aligned} (A \otimes B)(C \otimes D) &= AC \otimes BD \\ (A \otimes B)(u \otimes v) &= Au \otimes Bv \\ (u + v) \otimes w &= u \otimes w + v \otimes w \\ u \otimes (v + w) &= u \otimes v + u \otimes w \\ au \otimes bv &= ab(u \otimes v) \end{aligned}$$

  Thus for matrices,

  $$\begin{pmatrix} A & B \\ C & D \end{pmatrix} \otimes U = \begin{pmatrix} A \otimes U & B \otimes U \\ C \otimes U & D \otimes U \end{pmatrix},$$

  which specializes for scalars to

  $$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \otimes U = \begin{pmatrix} aU & bU \\ cU & dU \end{pmatrix}.$$

- The conjugate transpose distributes over tensor products:

$$(A \otimes B)^\dagger = A^\dagger \otimes B^\dagger.$$

- The tensor product of several matrices is unitary if and only if each one of the matrices is unitary up to a constant. Let $U = A_1 \otimes \ldots \otimes A_n$. Then U is unitary if $A_i^\dagger A_i = k_i I$ and $\prod_i k_i = 1$.

$$\begin{aligned} U^\dagger U &= (A_1^\dagger \otimes \ldots \otimes A_n^\dagger)(A_1 \otimes \ldots \otimes A_n) \\ &= A_1^\dagger A_1 \otimes \ldots \otimes A_n^\dagger A_n \\ &= k_1 I \otimes \ldots \otimes k_n I \\ &= I \end{aligned}$$

- Note that $\langle u \otimes v | w \otimes z \rangle = \langle u | w \rangle \langle v | z \rangle$. This implies that $\langle 0 \otimes u | 0 \otimes u \rangle = 0$, and therefore $0 \otimes u$ must be the zero vector of the tensor product Hilbert space.

  This in turn implies (reminds us?) that the tensor product space is actually the equivalence classes in a quotient space.

  In particular, if A and B are vector spaces, F is the free abelian group on $A \times B$, and K is the subgroup of F generated by all elements of the following forms (where $a, a_1, a_2 \in A, b, b_1, b_2 \in B, \alpha$ a scalar):

  1. $(a_1 + a_2, b) - (a_1, b) - (a_2, b)$

  2. $(a, b_1 + b_2) - (a, b_1) - (a, b_2)$

  3. $(\alpha a, b) - (a, \alpha b)$

  then $A \otimes B$ is the quotient space F/K.

Quantum bits (qubits) $\leftarrow$

- A quantum bit, or qubit, is a unit vector in a two dimensional complex vector space for which a particular orthonormal basis, denoted by $\{|0\rangle, |1\rangle\}$, has been fixed. It is important to notice that the basis vector $|0\rangle$ is NOT the zero vector of the vector space.

- For example, the basis $|0\rangle$ and $|1\rangle$ may correspond to the $|\uparrow\rangle$ and $|\rightarrow\rangle$ polarizations of a photon respectively, or to the polarizations $|\nearrow\rangle$ and $|\nwarrow\rangle$. Or $|0\rangle$ and $|1\rangle$ could correspond to the spin-up and spin-down states ($|\uparrow\rangle$ and $|\downarrow\rangle$) of an electron.

- For the purposes of quantum computing, the basis states $|0\rangle$ and $|1\rangle$ are taken to encode the classical bit values 0 and 1 respectively. Unlike classical bits however, qubits can be in a superposition of $|0\rangle$ and $|1\rangle$ such as $a|0\rangle + b|1\rangle$ where a and b are complex numbers such that $|a|^2 + |b|^2 = 1$. If such a superposition is measured with respect to the basis $\{|0\rangle, |1\rangle\}$, the probability that the measured value is $|0\rangle$ is $|a|^2$ and the probability that the measured value is $|1\rangle$ is $|b|^2$.

- Key properties of quantum bits:

  1. A qubit can be in a superposition state of 0 and 1.

  2. Measurement of a qubit in a superposition state will yield probabilistic results.

  3. Measurement of a qubit changes the state to the one measured.

  4. There is no transformation which exactly copies all qubits. This is known as the 'no cloning' principle. Interestingly, it is nonetheless possible to 'teleport' a quantum state, but in the process, the original quantum state is destroyed . . .

Magic

↞

”The Universe is full of magical things patiently waiting for our wits to grow sharper.”

-Eden Phillpotts

”Any sufficiently advanced technology is indistinguishable from magic.”

-Arthur C. Clarke

Entangled quantum states

←

- If we have available more than one (physical) qubit, we may be able to *entangle* them. The tensor product of the Hilbert spaces for the individual qubits is the appropriate model for these entangled systems.

- For example, if we have two qubits with bases $\{|0\rangle_1, |1\rangle_1\}$ and $\{|0\rangle_2, |1\rangle_2\}$ respectively, the tensor product space has the basis

  $$\{|0\rangle_1 \otimes |0\rangle_2, |0\rangle_1 \otimes |1\rangle_2, |1\rangle_1 \otimes |0\rangle_2, |1\rangle_1 \otimes |1\rangle_2\}.$$

  We can (conveniently) denote this basis as

  $$\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}.$$

- More generally, if we have n qubits to which we can apply common measurements, we will be working in the 2^n-dimensional Hilbert space with basis

  $\{|00\ldots00\rangle, |00\ldots01\rangle, \ldots, |11\ldots10\rangle, |11\ldots11\rangle\}$

- A typical quantum state for an n-qubit system is

$$\sum_{i=0}^{2^n-1} a_i|i\rangle$$

  where $a_i \in \mathbb{C}$, $\sum |a_i|^2 = 1$, and $\{|i\rangle\}$ is the basis, with (in our notation) i written as an n-bit binary number.

- A classical (macroscopic) physical object broken into pieces can be described and measured as separate components. An n-particle quantum system cannot always be described in terms of the states of its component pieces. For instance, the state $|00\rangle + |11\rangle$ cannot be decomposed into separate states of each of the two qubits in the form

  $$(a_1|0\rangle + b_1|1\rangle) \otimes (a_2|0\rangle + b_2|1\rangle).$$

  This is because

  $$(a_1|0\rangle + b_1|1\rangle) \otimes (a_2|0\rangle + b_2|1\rangle) =$$

  $$a_1a_2|00\rangle + a_1b_2|01\rangle + b_1a_2|10\rangle + b_1b_2|11\rangle$$

  and $a_1b_2 = 0$ implies that either $a_1a_2 = 0$ or $b_1b_2 = 0$. States which cannot be decomposed in this way are called entangled states. These are states that don't have classical counterparts, and for which our intuition is likely to fail.

- Particles are entangled if a measurement of one affects a measurement of the other. For example, the state $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ is entangled since the probability of measuring the first bit as $|0\rangle$ is 1/2 if the second bit has not been measured. However, if the second bit has been measured, the probability that the first bit is measured as $|0\rangle$ is either 1 or 0, depending on whether the second bit was measured as $|0\rangle$ or $|1\rangle$, respectively. On the other hand, the state $\frac{1}{\sqrt{2}}(|00\rangle + |01\rangle)$ is not entangled. Since $\frac{1}{\sqrt{2}}(|00\rangle + |01\rangle) = |0\rangle \otimes \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, any measurement of the first bit will yield $|0\rangle$ regardless of measurements of the second bit. Similarly, the second bit has a fifty-fifty chance of being measured as $|0\rangle$ regardless of measurements of the first bit. Note that entanglement in terms of particle measurement dependence is equivalent to the definition of entangled states as states that cannot be written as a tensor product of individual states.

Shocking

↞

“Anyone who is not shocked by quantum theory has not understood it.”

–Neils Bohr

“One is led to a new notion of unbroken wholeness which denies the classical analyzability of the world into separately and independently existing parts. The inseparable quantum interconnectedness of the whole universe is the fundamental reality.”

–David Bohm

“I don’t like it, and I’m sorry I ever had anything to do with it.”

–Erwin Schrodinger

Quantum computing ←

- This exponential growth in number of states, together with the ability to subject the entire space to transformations (either unitary dynamical evolution of the system, or a measurement projection into an eigenvector subspace), provides the foundation for quantum computing.

- An interesting (apparent) dilemma is the energetic costs/irreversability of classical computing. Since unitary transformations are invertible, quantum computations (except measurements) will all be reversible. Most classical boolean operations such as $b_1 \wedge b_2$, $b_1 \vee b_2$, and $b_1 \mathrel{\wedge\!\!\!\!\sim} b_2$ are irreversible, and therefore cannot directly be used as basic operations for quantum computers.

- The logical nand-gate ($b_1 \mathbin{\wedge\!\!\!\!\sim} b_2$) is sufficient to generate all the traditional boolean functions (e.g., $\sim b \equiv b \mathbin{\wedge\!\!\!\!\sim} b$). We are likely to end up looking for simple quantum gates that are similarly generic for quantum operations.

- In general, if we had enough time, we could simulate any quantum computation with a classical computer. The real potential value of quantum computers lies in speeding up computations. The critical questions are:

  1. How much can we speed up particular computations?

  2. Can we develop a practical implementation of a particular quantum computation?

  3. Can we build a physical implementation of a quantum computer?

4. Does the implementation allow us to carry out useful computations before decoherence interactions with the environment disturb the system too much?

5. Given the “no cloning” principle, can we develop quantum error detection/correction systems? In particular, we can’t just take measurements for error control since measurements have irreversible effects on quantum systems.

Simple quantum gates $\leftarrow$

- These are some examples of useful single-qubit quantum state transformations. Because of linearity, the transformations are fully specified by their effect on the basis vectors. The associated matrix is also shown.

$$\begin{array}{llll}
I : & |0\rangle \rightarrow |0\rangle & & \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \\
 & |1\rangle \rightarrow |1\rangle & & \\
\sigma_x : & |0\rangle \rightarrow |1\rangle & & \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \\
 & |1\rangle \rightarrow |0\rangle & & \\
\sigma_y : & |0\rangle \rightarrow |1\rangle & & \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \\
 & |1\rangle \rightarrow -|0\rangle & & \\
\sigma_z : & |0\rangle \rightarrow |0\rangle & & \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \\
 & |1\rangle \rightarrow -|1\rangle & &
\end{array}$$

  I is the identity transformation, σ_x is negation, σ_z is a phase shift operation, and $\sigma_y = \sigma_z \sigma_x$ is a combination of both. All these gates are unitary. For example

$$\sigma_y \sigma_y^\dagger = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} = I.$$

- Another important single-bit transformation is the Hadamard transformation defined by

  $$\begin{array}{rrcl} H: & |0\rangle & \rightarrow & \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \\ & |1\rangle & \rightarrow & \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle). \end{array}$$

  Applied to n bits each in the $|0\rangle$ state, the transformation generates a superposition of all 2^n possible states.

  $$\begin{array}{rl} & (H \otimes H \otimes \cdots \otimes H)|00\ldots 0\rangle \\ = & \frac{1}{\sqrt{2^n}}((|0\rangle + |1\rangle) \otimes \cdots \otimes (|0\rangle + |1\rangle)) \\ = & \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle. \end{array}$$

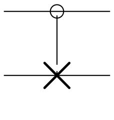
  The transformation acting on n bits is called the Walsh or Walsh-Hadamard transformation W.

- An important example of a two qubit gate is the controlled-not gate, C_{not}, which complements the second bit if the first bit is 1 and leaves the bit unchanged otherwise.

$$C_{not}: \begin{array}{ccc} |00\rangle & \rightarrow & |00\rangle \\ |01\rangle & \rightarrow & |01\rangle \\ |10\rangle & \rightarrow & |11\rangle \\ |11\rangle & \rightarrow & |10\rangle \end{array} \quad \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

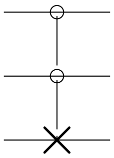
The transformation C_{not} is unitary since $C_{not}^{\dagger} = C_{not}$ and $C_{not}C_{not} = I$. The C_{not} gate cannot be decomposed into a tensor product of two single-bit transformations.

- It is useful to have graphical representations of quantum state transformations, especially when several transformations are combined. The controlled-not gate C_{not} is typically represented by a circuit of the form

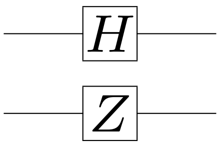


The open circle indicates the control bit, and the × indicates the conditional negation of the subject bit. In general there can be multiple control bits. Some authors use a solid circle to indicate negative control, in which the subject bit is toggled when the control bit is 0.

Similarly, the controlled-controlled-not, which negates the last bit of three if and only if the first two are both 1, has the following graphical representation.



Single bit operations are graphically represented by appropriately labelled boxes as shown.



- The bra/ket notation is useful in defining other unitary operations. Given two arbitrary unitary transformations U_1 and U_2, the “conditional” transformation $|0\rangle\langle 0| \otimes U_1 + |1\rangle\langle 1| \otimes U_2$ is also unitary. For example, the controlled-not gate can defined by

$$C_{not} = |0\rangle\langle 0| \otimes I + |1\rangle\langle 1| \otimes X.$$

- The three-bit controlled-controlled-not gate or Toffoli gate is also an instance of this conditional definition:

$$T = |0\rangle\langle 0| \otimes I \otimes I + |1\rangle\langle 1| \otimes C_{not}.$$

$$\begin{array}{rlcl} T: & |000\rangle & \rightarrow & |000\rangle \\ & |001\rangle & \rightarrow & |001\rangle \\ & |010\rangle & \rightarrow & |010\rangle \\ & |011\rangle & \rightarrow & |011\rangle \\ & |100\rangle & \rightarrow & |100\rangle \\ & |101\rangle & \rightarrow & |101\rangle \\ & |110\rangle & \rightarrow & |111\rangle \\ & |111\rangle & \rightarrow & |110\rangle \end{array}$$

T can be used to construct a complete set of the classical boolean connectives and thus general combinatory circuits since it can be used to construct the not and and operators in the following way:

$$\begin{array}{rcl} T|1,1,x\rangle & = & |1,1,\sim x\rangle \\ T|x,y,0\rangle & = & |x,y,x \wedge y\rangle \end{array}$$

Tractability of computation

←

- We can generally categorize computational algorithms according to how the resources needed for execution of the algorithm increase as we increase the size of the input. Typical resources are time and (storage) space. In different contexts, we may be interested in worst-case or average-case performance of the algorithm. For theoretical purposes, we will typically be interested in large input sets . . .

- The hope of quantum computing is that problems that are difficult or impossible for classical computers to solve can be handled by quantum computers.

- A standard mechanism for comparing the growth of functions with domain $\mathbb{N}$ is “big-Oh.” One way of defining this notion is to associate each function with a set of functions. We can then compare algorithms by looking at their “big-Oh” categories.

- Given a function f, we define $O(f)$ by:

$$g \in O(f) \iff$$

$$\text{there exist } c > 0 \text{ and } N \geq 0 \text{ such that}$$
$$|g(n)| \leq c|f(n)| \text{ for all } n \geq N.$$

- We further define $\theta(f)$ by:
  $g \in \theta(f)$ iff $g \in O(f)$ and $f \in O(g)$.

- In general we will consider the run-time of algorithms in terms of the growth of the number of elementary computer operations as a function of the number of bits in the (encoded) input. Some important categories – an algorithm's run-time f is:

  1. Logarithmic if $f \in \theta(\log(n))$.

  2. Linear if $f \in \theta(n)$.

  3. Quadratic if $f \in \theta(n^2)$.

  4. Polynomial if $f \in \theta(P(n))$ for some polynomial $P(n)$.

  5. Exponential if $f \in \theta(b^n)$ for some constant $b > 1$.

  6. Factorial if $f \in \theta(n!)$.

- Typically we say that a problem is *tractable* if (we know) there exists an algorithm whose run-time is (at worst) polynomial that solves the problem. Otherwise, we call the problem *intractable*.

- There are many problems which have the interesting property that if someone (an oracle?) provides you with a solution to the problem, you can tell in polynomial time whether what they provided you actually is a solution. Problems with this property are called Non-deterministically Polynomial, or NP, problems. One way to think about this property is to imagine that we have arbitrarily many machines available. We let each machine work on one possible solution, and whichever machine finds the (a) solution lets us know.

- There are some even more interesting NP problems which are universal for the class of NP problems. These are called NP-complete problems. A problem S is NP-complete if S is NP and, there exists a polynomial time algorithm that allows us to translate any NP problem into an instance of S. If we could find a polynomial time algorithm to solve a single NP-complete problem, we would then have a polynomial time solution for each NP problem.

- Some examples:

  1. Factoring a number is NP. First, we recognize that if M is the number we want to factor, then the input size m is approximately log(M) (that is, the input size is the number of digits in the number). The elementary school algorithm (try dividing by each number less than $\sqrt{M}$) has run-time approximately $10^{\frac{m}{2}}$, which is exponential in the number of digits. On the other hand, if someone hands you two numbers they claim are factors of M, you can check by multiplying, which takes on the order of m^2 operations.

     It is worth noting that there is a polynomial time algorithm to determine whether or not a number is prime, but for composite numbers, this algorithm does not provide a factorization. Factoring is a particularly important example because various encryption algorithms such as RSA (used in the PGP software) depend for their security on the difficulty of factoring numbers with several hundred digits.

2. Satisfiability of a boolean expression is NP-complete. Suppose we have n boolean variables $\{b_1, b_2, \ldots, b_n\}$ (each with the possible values 0 and 1). We can form a general boolean expression from these variables and their negations:

$$f(b_1, b_2, \ldots, b_n) = \bigwedge_k (\bigvee_{i,j \leq n} (b_i, \sim b_j)).$$

A solution to such a problem is an assignment of values 0 or 1 to each of the b_i such that $f(b_1, b_2, \ldots, b_n)$ =1. There are 2^n possible assignments of values. We can check an individual possible solution in polynomial time, but there are exponentially many possibilities to check. If we could develop a feasible quantum computation for this problem, we would in some sense resolve the traditional P$\overset{?}{=}$NP problem . . .

3. The discrete Fourier transform of a sequence $\overrightarrow{a} = \langle a_j \rangle_{j=0}^{q-1}$ is the sequence $\overrightarrow{A} = \langle A_k \rangle_{k=0}^{q-1}$ where

$$A_k = \frac{1}{\sqrt{q}} \sum_{j=0}^{q-1} a_j e^{\frac{2\pi ijk}{q}}$$

One way to think about this is that $\overrightarrow{A} = F\overrightarrow{a}$ where the linear transformation F is given by:

$$[F]_{j,k} = \frac{1}{\sqrt{q}} e^{\frac{2\pi ijk}{q}}$$

Note that the inverse of F is $F^\dagger$ – that is,

$$[F^{-1}]_{k,j} = \frac{1}{\sqrt{q}} e^{-\frac{2\pi ijk}{q}}.$$

Suggestively, this says that the discrete Fourier transform is a unitary operation.

The action of this transformation on a vector of dimension q looks as though it would take the q^2 operations of matrix multiplication, but there is enough structure that the classical fast Fourier transform algorithm can be done in $q \log(q)$ operations.

The corresponding quantum Fourier transform U_{QFT} with base 2^n is defined by

$$U_{QFT} : |x\rangle \mapsto \frac{1}{\sqrt{2^n}} \sum_{c=0}^{2^n-1} e^{\frac{2\pi icx}{2^n}} |c\rangle.$$

We will see that this can be accomplished in approximately n^2 operations rather than $n2^n$. This is an exponential speed-up of the process.

Factoring

←

- The quantum algorithm which has probably done the most for popularizing quantum computation is Shor's factoring algorithm. As noted above, a fast algorithm for factoring numbers with several hundred digits would invalidate some of the most widely used encryption systems. Shor's algorithm provides theoretical evidence for such an algorithm, waiting only for a practical physical realization.

- The general approach used by Shor is based on a classical probabilistic method for factoring. The classical algorithm is exponential in the number of digits – Shor's is (quantum) polynomial.

- Outline of Shor's algorithm for factoring a number M:

  1. Choose an integer $1 < y < M$ arbitrarily. If y is not relatively prime to M, we've found a factor of M. Otherwise apply the rest of the algorithm.

  2. Let n be such that $M^2 \leq 2^n < 2M^2$. We begin with n qubits, each in state $|0\rangle$. We now apply the Walsh transformation W to superpose all states:
$$\sum_{a=0}^{2^n-1} |0\rangle \overset{W}{\longmapsto} \frac{1}{\sqrt{2^n}} \sum_{a=0}^{2^n-1} |a\rangle.$$

  3. Apply a transformation which implements raising to powers (mod M):
$$\frac{1}{\sqrt{2^n}} \sum_{a=0}^{2^n-1} |a\rangle \mapsto \frac{1}{\sqrt{2^n}} \sum_{a=0}^{2^n-1} |a, f(a)\rangle$$
where $f(a) = y^a$ (mod M).

4. Measure to find a state whose amplitude has the same period as f.

5. Apply a quantum Fourier transform to invert the frequency.

6. Extract the period, which we expect to be the order of $y \pmod{M}$.

7. Find a factor of M.

   When our estimate for the period, q, is even, we use the Euclidean algorithm to efficiently check whether either $y^{q/2} + 1$ or $y^{q/2} - 1$ has a non-trivial common factor with M.

8. Repeat the algorithm, if necessary.

- Here's another version of the outline of Shor's algorithm for factoring

  We begin with 2 n-qubit registers. Apply the Walsh transformation on the first to give a uniform superposition of states:

$$|\vec{0}\rangle \otimes |\vec{0}\rangle \Rightarrow \frac{1}{\sqrt{Q}} \sum_{l=0}^{Q-1} |l\rangle \otimes | \vec{0}\rangle$$

  Apply a transformation which computes $y^l \ mod \ N$:

$$\frac{1}{\sqrt{Q}} \sum_{l=0}^{Q-1} |l\rangle \otimes |y^l modN\rangle$$

  Measure the second register:

$$\frac{1}{\sqrt{A}} \sum_{l=0|y^l=y^{l_0}}^{Q-1} |l\rangle \otimes |y^{l_0}\rangle =$$

$$\frac{1}{\sqrt{A}} \sum_{j=0}^{A-1} |jr + l_0\rangle \otimes |y^{l_0}\rangle$$

Apply the quantum Fourier transform over Z_Q on the first register:

$$\frac{1}{\sqrt{Q}}\sum_{k=0}^{Q-1}\left(\frac{1}{\sqrt{A}}\sum_{j=0}^{A-1}e^{2\pi i(jr+l_0)k/Q}\right)|k\rangle\otimes|y^{l_0}\rangle$$

Measure the first register. Let k_1 be the outcome. Approximate the fraction $\frac{k_1}{Q}$ by a fraction with denominator smaller than N. If the denominator d doesn't satisfy $y^d = 1 \bmod N$, throw it away, else call the denominator r_1.

Repeat all previous steps poly(log(N)) times to get r_1, r_2, ...

Output the minimal r.

Notes on factoring $\leftarrow$

- To factor a number M, we choose a number $y < M$ with $gcd(y, M) = 1$. We then find r, the order of y in the multiplicative group (mod M). If r is even, then $(y^{r/2} + 1)(y^{r/2} - 1) = (y^r - 1) \equiv 0 \pmod{M}$. Then $gcd(y^r - 1, M)$ is a non-trivial factor of M except when r is odd or $y^{r/2} \equiv -1 \pmod{M}$. This procedure produces a non-trivial factor of M with probability at least $1 - 1/2^{k-1}$, where k is the number of distinct odd prime factors of M. If we don't get a factor, we can choose a new y and repeat the process. By repeating the process, we can make our likelihood of success as close to one as we like. Note that if M is even, finding a factor is easy; if M is a power of a prime, there are other fast classical methods of factoring which we can use on M before we start this process.

- We want to find the period of the function $f(a) = y^a \pmod{M}$. We do that by measuring to find a state whose amplitude has the same period as f.

  We measure the qubits of the state obtained from encoding $f(a)$. A random value u is obtained. We don't actually use the value u; only the effect the measurement has on our set of superpositions is of interest. This measurement projects the state space onto the subspace compatible with the measured value, so the state after measurement is

$$C \sum_a g(a) |a, u\rangle,$$

  for some scale factor C where

$$g(a) = \begin{cases} 1 & \text{if } f(a) = u \\ 0 & \text{otherwise} \end{cases}$$

  Note that the a's that actually appear in the sum, those with $g(a) \neq 0$, differ from each other by multiples of the period, and thus $g(a)$ is the function we are looking for. If we could just measure two successive a's in the sum, we would have the period. Unfortunately the quantum world permits only one measurement.

- Shor's method uses a quantum version of the Fourier transform to find the period of the function y^a (mod M). We apply the quantum Fourier transform to the state obtained by the measurement.

  $$\sum_a g(a)|a\rangle \overset{QFT}{\longmapsto} \sum_c G(c)|c\rangle$$

  Standard Fourier analysis tells us that when the period r of $g(a)$ is a power of two, the result of the quantum Fourier transform is

  $$C' \sum_j \rho_j |j \frac{2^n}{r}\rangle$$

where $|\rho_j| = 1$. When the period r does not divide 2^n, the transform approximates the exact case so most of the amplitude is attached to integers close to multiples of $\frac{2^n}{r}$.

- In order for Shor's factoring algorithm to be a polynomial algorithm, the quantum Fourier transform must be efficiently computable. Shor developed a quantum Fourier transform construction with base 2^n using only $\frac{n(n+1)}{2}$ gates. The construction makes use of two types of gates. One is a gate to perform the Hadamard transformation H. We will denote by H_j the Hadamard transformation applied to the jth bit. The other type of gate performs transformations of the form

$$S_{j,k} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{i\theta_{k-j}} \end{pmatrix}$$

where $\theta_{k-j} = \pi/2^{k-j}$, which acts on the kth element, depending on the value of the jth element. Think of this as acting on the basis $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$ . . .

The quantum Fourier transform is given by

$$H_0 S_{0,1} \ldots S_{0,n-1} H_1 \ldots$$

$$H_{n-3} S_{n-3,n-2} S_{n-3,n-1} H_{n-2} S_{n-2,n-1} H_{n-1}.$$

This actually produces the reverse of the Fourier transform, so it typically will be followed by a bit reversal transformation.

- There is a second piece of Shor's algorithm which must be accomplished in polynomial time. We need to extract (using the QFT) the period of the function $a \mapsto (y^a)$ (mod M). We must transform as:

  $$\frac{1}{\sqrt{2^n}} \sum_{a=0}^{2^n-1} |a\rangle \mapsto \frac{1}{\sqrt{2^n}} \sum_{a=0}^{2^n-1} |a, y^a(modM)\rangle$$

  We want to develop a transformation which computes the function $f_{y,M}(a) = y^a$ (mod M). First, we write y^a as $y^a = y^{2^0 a_0} \cdot y^{2^1 a_1} \cdot \ldots y^{2^{m-1} a_{m-1}}$, where m is the number of digits in the binary expansion of M. Then, modular exponentiation can be computed by initializing the result register to $|1\rangle$, and successively effecting m multiplications by y^{2^i} (mod M), depending on the value of the qubit $|a_i\rangle$.

If $a_i = 1$, we want the operation

$$|y^{2^0 a_0 + \ldots 2^{i-1} a_{i-1}}, 0\rangle \mapsto$$

$$|y^{2^0 a_0 + \ldots 2^{i-1} a_{i-1}}, y^{2^0 a_0 + \ldots 2^{i-1} a_{i-1}} \cdot y^{2^i}\rangle$$

to be performed; otherwise, when $a_i = 0$ we just require

$$|y^{2^0 a_0 + \ldots 2^{i-1} a_{i-1}}, 0\rangle \mapsto$$

$$|y^{2^0 a_0 + \ldots 2^{i-1} a_{i-1}}, y^{2^0 a_0 + \ldots 2^{i-1} a_{i-1}}\rangle.$$

Note that in both cases the result can be written as $|y^{2^0 a_0 + \ldots 2^{i-1} a_{i-1}}, y^{2^0 a_0 + \ldots 2^i a_i}\rangle$.

- To extract the period, we measure the state in the standard basis for quantum computation, and call the result v. In the case where the period happens to be a power of 2 so that the quantum Fourier transform gives exactly multiples of the scaled frequency, the period is easy to extract. In this case, $v = j\frac{2^n}{r}$ for some j. Most of the time j and r will be relatively prime, in which case reducing the fraction $\frac{v}{2^n}$ to its lowest terms will yield a fraction whose denominator q is the period r. The fact that in general the quantum Fourier transform only gives approximately multiples of the scaled frequency complicates the extraction of the period from the measurement. When the period is not a power of 2, a good guess for the period can be obtained using the continued fraction expansion of $\frac{v}{2^n}$.

- Various things could have gone wrong so that this process does not yield a factor of M:

1. The value v was not close enough to a multiple of $\frac{2^n}{r}$.

2. The period r and the multiplier j could have had a common factor so that the denominator q was actually a factor of the period, rather than the period itself.

3. We find M as M's factor.

4. The period of $f(a) = y^a \pmod{M}$ is odd.

A few repetitions of this algorithm yields a factor of M with high probability.

Quantum algorithms for satisfiability

←

- Various approaches have been developed which provide hope that the NP-complete boolean satisfiability problem can be solved in polynomial time. It is not clear that any of the published techniques will be effective. Some of the methods seem to require either exponential space/hardware (e.g., bulk spin resonance via NMR) or exponential measurement precision. This is a very active area of current research.

  One algorithm which has been well analyzed is Grover's search algorithm. It gives quadratic speedup of solving satisfiability, but in its general form can do no better than that, and hence does not give the exponential speedup needed to get $P = NP$.

- Following is an outline of Grover's general search algorithm. If $P(x)$ is a boolean function for $0 \leq x < N$, classical search algorithms take on the order of $\frac{N}{2}$ operations to find an item x_0 for which $P(x_0) = 1$. Grover's algorithm takes on the order of $\sqrt{N}$ operations. Grover's algorithm has been shown to be optimal for the general search problem. This is not an exponential speedup, but it is an improvement over the classical algorithms. However, problems such as satisfiability have additional structure which can make them easier to solve.

- Grover's algorithm consists of the following steps:

  1. Let n be such that $2^n \geq N$, and prepare a register containing a superposition of all $x_i \in [0 \ldots 2^n - 1]$.

2. Apply a unitary transformation that computes $P(x_i)$ on this register:

   $$U_P : \frac{1}{\sqrt{2^n}} \sum_{x=0}^{n-1} |x, 0\rangle \rightarrow \frac{1}{\sqrt{2^n}} \sum_{x=0}^{n-1} |x, P(x)\rangle.$$

   For any x_0 such that $P(x_0)$ is true, $|x_0, 1\rangle$ will be part of the resulting superposition, but since its amplitude is $\frac{1}{\sqrt{2^n}}$, the probability that a measurement produces x_0 is only 2^{-n}.

3. Change amplitude a_j to $-a_j$ for all x_j such that $P(x_j) = 1$.

4. Apply inversion about the average to increase amplitude of x_j with $P(x_j) = 1$ and decrease other amplitudes.

5. Repeat steps 2 through 4 $\frac{\pi}{4}\sqrt{2^n}$ times.

6. Measure the last qubit of the quantum state, representing $P(x)$. Because of the amplitude change, there is a high

probability that the result will be 1. If this is the case, the measurement has projected the state onto the subspace $\frac{1}{\sqrt{2^k}}\sum_{i=1}^{k}|x_i, 1\rangle$ where k is the number of solutions. Further measurement of the remaining bits will provide one of these solutions.

- An interesting feature of this algorithm is that repeating steps 2 through 4 a total of $\frac{\pi}{4}\sqrt{2^n}$ times is optimal. In particular, if the process is repeated more times, the probability of a successful measurement decreases back toward zero . . .

- An alternative approach builds the unitary transformation for the boolean expression, applies the transformation to molecules in solution, then uses bulk spin resonance analysis via NMR to measure the expected values of the spins, and thus solves the satisfiability problem. However, realistic implementations seem to require an exponentially large NMR sample.

- The general estimate is that if n is the number of qubits, and M is the number of molecules in the sample, then $n2^n < M$. For a typical sample, $M \approx 10^{24} \approx 2^{80}$ and so $n < 74$. For an upper limit, a reasonable estimate of the number of elementary particles in the accessible universe is $\approx 10^{80} \approx 2^{265}$ which corresponds with ≈ 256 qubits . . .

Possibilities for physical implementation ←

- Implementations of quantum computers will be a difficult experimental challenge. Quantum computer equipment must satisfy a variety of constraints: (1) the qubits must interact very weakly with their environment to minimize decoherence and preserve their superpositions, (2) the qubits must interact very strongly with one another for the logic gates and information transfer to be effective, and (3) the initialization and readout of states must be efficient. Not many known physical systems can satisfy these requirements, although there are some possibilities.

- A collection of charged ions held in an electromagnetic trap is one possibility. Each atom stores a qubit of information in a pair of internal electron levels. Each atom's levels are protected from environmental influences. Scaling to larger numbers of qubits should be able to be done by adding more atoms to the collection. When appropriate laser radiation is applied to the atoms, only one of the two internal states fluoresces. This allows detection of the state of each qubit. The atoms are coupled by virtue of their mutual Coulomb repulsion. Experimental development of trapped ion quantum computation is at the level of single-ion and two-ion qubit systems. Extensions to larger numbers of trapped ions has been difficult, but there do not seem to be impossible theoretical limits to scaling.

- Another system which could be developed into a quantum computer is a single molecule, in which nuclear spins of individual atoms represent qubits. This is the basis of the NMR technique mentioned above. The spins can be manipulated, initialized, and measured. For example, the carbon and hydrogen nuclei in a chloroform molecule can be used to represent two qubits. Applying a radio-frequency pulse to the hydrogen nucleus addresses that qubit and causes it to rotate from a $|0\rangle$ state to a superposition $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ state. Interactions through chemical bonds allow multiple-qubit logic to be performed. However, it is difficult to find molecules with more than 10 spins in them and with a large coupling constant between every pair of spins . . .

Decoherence and error correction

←

- Decoherence in general arises from interactions with the environment, which typically has the effect of measuring the system and thus collapsing a quantum computation. In addition, we have to be careful about leaving temporary qubits floating around. We can expect them to be entangled with the rest of the system, and thus an observation of the “dust” left behind by intermediate computations could effect a measurement of the system, invalidating later stages. Thus, one emphasis in research on quantum computation has been on how to efficiently avoid leaving any garbage floating about.

- As noted above, error detection/correction is difficult in the quantum environment since we cannot reliably clone an arbitrary qubit. Further, any intermediate measurement of the system for error control is likely to invalidate our computation. There are, however, approaches using polarization encoding schemes for error control.

Prospects

←

- The history of quantum mechanical algorithms is very brief. There are two main approaches that have resulted in descriptions of efficient quantum computational algorithms: the first is estimates of periodicity that resulted in the factorization algorithm, and the second is amplitude amplification that has led to Grover's quantum search and related algorithms.

- Over the past 70 or 80 years, physicists have observed various quantum mechanical phenomena that lead to puzzling and even apparently paradoxical results. Most of these still remain to be investigated from a quantum computing perspective.

- One interesting question is how slight difference in the laws of quantum mechanics might affect these issues. Some interesting work by Abrams et al. shows that if there was even the slightest amount of nonlinearity in quantum mechanics, it would be possible to modify the amplitude amplification scheme of Grover's quantum search algorithm to obtain an efficient algorithm solving the NP-complete satisfiability problem. However, most people believe that such nonlinearity probably does not exist because it would also lead to faster-than-light communication, noncausality, and other violations of fundamental physical principles . . .

Finis

↞

“Nature uses only the longest threads to weave her patterns, so that each small piece of her fabric reveals the organization of the entire tapestry.”

– Richard Feynman

To top ←

References

[1] Abrams D S and Lloyd S, Non-Linear Quantum Mechanics implies Polynomial Time solution for NP-complete and #P problems, http://xxx.lanl.gov/abs/quant-ph/9801041

[2] Aharonov D, Beckman D, Chuang I and Nielsen M, What Makes Quantum Computers Powerful? http://wwwcas.phys.unm.edu/m̃nielsen/science.html

[3] Aharonov, D., Quantum Computation, Annual Reviews of Computational Physics VI, Edited by Dietrich Stauffer, World Scientific, 1998

[4] Barenco A A universal two-bit gate for quantum computation, *Proc. R. Soc. Lond. A* **449** 679–683, 1995

[5] Barenco A, Deutsch D, Ekert E and Jozsa R, Conditional quantum dynamics and quantum gates, *Phys. Rev. Lett.* **74** 4083–4086, 1995

[6] Barenco A, Bennett C H, Cleve R, DiVincenzo D P, Margolus N, Shor P, Sleator T, Smolin J A and Weinfurter H, Elementary gates for quantum computation, *Phys. Rev. A* **52**, 3457–3467, 1995

[7] Bell J S On the Einstein-Podolsky-Rosen paradox, *Physics* **1** 195–200, 1964

[8] Bell J S On the problem of hidden variables in quantum theory, Rev. Mod. Phys. **38** 447-52, 1966 *Speakable and unspeakable in quantum mechanics* 1987 (Cambridge University Press)

[9] Bennett C H, Brassard G, Crépeau C, Jozsa R, Peres A and Wootters W K Teleporting an unknown quantum state via dual classical and Einstein-Podolsky-Rosen channels, *Phys. Rev. Lett.* **70** 1895–1898, 1993

[10] Bennett C H, DiVincenzo D P, Smolin J A and Wootters W K Mixed state entanglement and quantum error correction, *Phys. Rev. A* **54** 3825, 1996

[11] Bennett C H, Bernstein E, Brassard G and Vazirani U Strengths and Weaknesses of quantum computing, *SIAM Journal of Computation* **26** 5 pp 1510–1523 October, 1997

[12] Boyer M, Brassard G, Hoyer P and Tapp A, Tight bounds on quantum searching, in *Fortsch.Phys.* 46, (1998) pp. 493-506

[13] Brassard G, Searching a quantum phone book, **Science 275** 627-628 1997

[14] Calderbank A R and Shor P W, Good quantum error-correcting codes exist, *Phys. Rev. A* **54** 1098-1105, 1996

[15] Chuang I L, Laflamme R, Shor P W and Zurek W H, Quantum computers, factoring, and decoherence, *Science* **270** 1633–1635, 1995

[16] Chuang I L, Laflamme R and Paz J P, Effects of Loss and Decoherence on a Simple Quantum Computer, http://xxx.lanl.gov/abs/quant-ph/9602018

[17] Clausen M, Fast Generalized Fourier transforms, *Theoret. Comput. Sci.* **56** 55–63 1989

[18] Coppersmith D, An approximate Fourier transform useful in quantum factoring, IBM Research Report RC 19642, 1994

[19] Cormen T, Leiserson C and Rivest R, *Introduction to Algorithms*, (pp 776–800 for FFT, 837–844 for primality test, 812 for extended Euclid algorithm, 834–836 for RSA cryptosystem) MIT press, 1990

[20] Cory D G, Fahmy A F, and Havel T F, Nuclear magnetic resonance spectroscopy: an experimentally accessible paradigm for quantum computing, in *Proc. of the 4th Workshop on Physics and Computation* (Complex Systems Institute, Boston, New England) 1996

[21] Deutsch, D., The Fabric of Reality, Penguin Books Ltd, Harmondsworth, Middlesex, England, 1997

[22] Deutsch D, Quantum theory, the Church-Turing principle and the universal quantum computer, In *Proc. Roy. Soc. Lond.* A **400** 97-117, 1985

[23] Deutsch D, Quantum computational networks, In *Proc. Roy. Soc. Lond.* A **425** 73-90, 1989

[24] Deutsch D and Jozsa R, Rapid solution of problems by quantum computation, In *Proc. Roy. Soc. Lond* A **439** 553-558, 1992

[25] Deutsch D, Barenco A and Ekert A, Universality in quantum computation, In *Proc. R. Soc. Lond.* A **449** 669-677, 1995

[26] DiVincenzo D P, Two-bit gates are universal for quantum computation, *Phys. Rev. A* **51** 1015-1022 1995

[27] DiVincenzo D P, Quantum computation, *Science* **270** 255-261 1995

[28] Einstein A, Rosen N and Podolsky B, *Phys. Rev.* **47**, 777 1935

[29] Ekert A and Jozsa R Quantum computation and Shor's factoring algorithm, *Rev. Mod. Phys.* **68** 733 1996

[30] Feynman R P Simulating physics with computers, In *Int. J. Theor. Phys.* **21** 467-488, 1982

[31] Feynman R P, Quantum mechanical computers, In *Found. of Phys.* **16** 507-531, 1986 see also Optics News February 1985, 11-20.

[32] R. Feynman, Feynman lectures on computation, 1996.

[33] Garey M R and Johnson D S, Computers and Intractability, published by Freeman and Company, New York, 1979

[34] Gershenfeld N A and Chuang I L Bulk spin-resonance quantum computation, *Science*, 275:350–356, 1997.

[35] Grover L K, Quantum mechanics helps in searching for a needle in a haystack, Phys. Rev. Lett. **79**, 325-328 1997 and the original STOC paper: A fast quantum mechanical algorithm for database search *Proc. of the 28th Annual ACM Symposium on Theory of Computing (STOC)* 212–221, 1996

[36] Grover L K, A framework for fast quantum mechanical algorithms, http://xxx.lanl.gov/abs/quant-ph/9711043

[37] Grover L K, Quantum computers can search arbitrarily large databases by a single query in *Phys. Rev. Let.* **79** 23, 4709–4712, 1997

[38] Grover L K, A fast quantum mechanical algorithm for estimating the median, http://xxx.lanl.gov/abs/quant-ph/9607024

[39] Hagley E et. al, Generation of Einstein Podolsky Rosen pairs of atoms, *Phys. Rev. Lett,* **79**, 1–5, 1997

[40] Hamming R W 1986 *Coding and information theory*, 2nd ed, (Prentice-Hall, Englewood Cliffs)

[41] Hardy G H and Wright E M 1979 *An introduction to the theory of numbers* (Clarendon Press, Oxford)

[42] Haroche S and Raimond J-M 1996 Quantum computing: dream or nightmare? Phys. Today August 51-52

[43] Hodges A 1983 *Alan Turing: the enigma* (Vintage, London)

[44] Hungerford T W, 1974 *Algebra* (Springer-Verlag, New York)

[45] A. J. Jones, M. Mosca and R. H. Hansen, Implementation of a Quantum Search Algorithm on a Nuclear Magnetic Resonance Quantum Computer, in *Nature* 393 (1998) 344-346, and see also A. J. Jones and M. Mosca, Implementation of a Quantum Algorithm to Solve Deutsch's Problem on a Nuclear Magnetic Resonance Quantum Computer, in *J. Chem. Phys.* 109 (1998) 1648-1653

[46] Knill E and Laflamme R 1997 A theory of quantum error-correcting codes, Phys. Rev. A **55** 900-911

[47] Knill E, Laflamme R and Zurek W H 1997 Resilient quantum computation: error models and thresholds http://xxx.lanl.gov/abs/quant-ph/9702058

[48] Knuth D E 1981 *The Art of Computer Programming, Vol. 2: Seminumerical Algorithms*, 2nd ed (Addison-Wesley).

[49] Lipton R J, Using DNA to solve NP-complete problems. *Science*, **268** 542–545, Apr. 28, 1995

[50] Lloyd S 1995 Almost any quantum logic gate is universal, Phys. Rev. Lett. **75**, 346-349

[51] Margolus N 1990 Parallel Quantum Computation, in *Complexity, Entropy and the Physics of Information, Santa Fe Institute Studies in the Sciences of Complexity,* vol VIII p. 273 ed Zurek W H (Addison-Wesley)

[52] von Neumann, Probabilistic logic and the synthesis of reliable organisms from unreliable components, in *automata studies(Shanon,McCarthy eds)*, 1956

[53] Papadimitriou C H, Computational Complexity, Addison-Wesley, 1994

[54] Peres A 1993 *Quantum theory: concepts and methods* (Kluwer Academic Press, Dordrecht)

[55] Preskill J 1997 Fault tolerant quantum computation, to appear in *Introduction to Quantum Computation*, edited by H.-K. Lo, S. Popescu, and T. P. Spiller http://xxx.lanl.gov/abs/quant-ph/9712048

[56] Preskill J, Kitaev A, Course notes for Physics 229, Fall 1998, Caltech Univ., http://www.theory.caltech.edu/people/preskill/ph229

[57] Rieffel E, Polak W An Introduction to Quantum Computing for Non-Physicists http://xxx.lanl.gov/abs/quant-ph/9809016

[58] Rivest R, Shamir A and Adleman L 1979 On digital signatures and public-key cryptosystems, MIT Laboratory for Computer Science, Technical Report, MIT/LCS/TR-212

[59] J.J.Saqurai Modern Quantum Mechanics, revised edition. Addison Wesley, 1994

[60] Shannon C E 1948 A mathematical theory of communication Bell Syst. Tech. J. **27** 379; also p. 623

[61] Shor P W, Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer, *SIAM J. Comp.*, **26**, No. 5, pp 1484–1509, October 1997

[62] Steane A M, Error correcting codes in quantum theory, Phys. Rev. Lett. **77** 793-797, 1996, Simple quantum error-correcting codes, Phys. Rev. A **54**, 4741-4751, 1996, Quantum Reed-Muller codes, submitted to IEEE Trans. Inf. Theory (preprint in *LANL e-print* quant-ph/9608026, http://xxx.lanl.gov) Active stabilization, quantum computation, and quantum state synthesis, Phys. Rev. Lett. **78**, 2252-2255, 1997

[63] Steane A, Quantum Computation, Reports on Progress in Physics 61 (1998) 117, preprint in http://xxx.lanl.gov/abs/quant-ph/9708022

[64] Toffoli T 1980 Reversible computing, in *Automata, Languages and Programming*, Seventh Colloquium, Lecture Notes in Computer Science, Vol. 84, de Bakker J W and van Leeuwen J, eds, (Springer) 632-644

[65] Turing A M 1936 On computable numbers, with an application to the Entschneidungsproblem, Proc. Lond. Math. Soc. Ser. 2 **42**, 230 ; see also Proc. Lond. Math. Soc. Ser. 2 **43**, 544

[66] Wheeler J A and Zurek W H, eds, 1983 *Quantum theory and measurement* (Princeton Univ. Press, Princeton, NJ)

[67] Wootters W K and Zurek W H 1982 A single quantum cannot be cloned, Nature **299**, 802

[68] Zalka C, Grover's quantum searching algorithm is optimal, http://xxx.lanl.gov/abs/quant-ph/9711070

[69] Zurek W H, Decoherence and the transition from quantum to classical, Physics Today 44(10), October, 1991 36–44.

To top ←

On-line references

←

Some of the references listed above are available on line. They are listed again here for easy access:

Abrams D S and Lloyd S, Non-Linear Quantum Mechanics implies Polynomial Time solution for NP-complete and #P problems, http://xxx.lanl.gov/abs/quant-ph/9801041

Aharonov D, Quantum Computation, http://xxx.lanl.gov/abs/quant-ph/9812037

Chuang I L, Laflamme R and Paz J P, Effects of Loss and Decoherence on a Simple Quantum Computer, http://xxx.lanl.gov/abs/quant-ph/9602018

Grover L K, A framework for fast quantum mechanical algorithms, http://xxx.lanl.gov/abs/quant-ph/9711043

Grover L K, A fast quantum mechanical algorithm for estimating the median, http://xxx.lanl.gov/abs/quant-ph/9607024

Knill E, Laflamme R and Zurek W H 1997 Resilient quantum computation: error models and thresholds http://xxx.lanl.gov/abs/quant-ph/9702058

Preskill J 1997 Fault tolerant quantum computation, to appear in *Introduction to Quantum Computation*, edited by H.-K. Lo, S. Popescu, and T. P. Spiller http://xxx.lanl.gov/abs/quant-ph/9712048

Preskill J, Kitaev A, Course notes for Physics 229, Fall 1998, Caltech Univ., http://www.theory.caltech.edu/people/preskill/ph229

Rieffel E, Polak W An Introduction to Quantum Computing for Non-Physicists http://xxx.lanl.gov/abs/quant-ph/9809016

Steane A, Quantum Computation, Reports on Progress in Physics 61 (1998) 117, http://xxx.lanl.gov/abs/quant-ph/9708022

Zalka C, Grover's quantum searching algorithm is optimal, http://xxx.lanl.gov/abs/quant-ph/9711070

To top ←