Quantum Computation

Concepts and Prospects

Apoorva Patel

Centre for High Energy Physics and

Supercomputer Education and Research Centre

Indian Institute of Science, Bangalore

5 March 2010, Conscientia 2010, IIST, Thiruvananthapuram

What is a quantum computer?

It is a computer whose elementary hardware components work according to the laws of quantum mechanics.
(The hardware components of classical digital computers work according to the laws of electronic circuits.)

What is a quantum computer?

It is a computer whose elementary hardware components work according to the laws of quantum mechanics.
(The hardware components of classical digital computers work according to the laws of electronic circuits.)

There is a lot more to computation than Boolean algebra!
In going from classical to quantum computers, the concept of what is computable and what is not does not change, but the criteria of computational efficiency do.

What is a quantum computer?

It is a computer whose elementary hardware components work according to the laws of quantum mechanics.
(The hardware components of classical digital computers work according to the laws of electronic circuits.)

There is a lot more to computation than Boolean algebra!
In going from classical to quantum computers, the concept of what is computable and what is not does not change, but the criteria of computational efficiency do.

Present Status

Laws of quantum mechanics are precisely known.
Theoretical foundation of the subject is clear.
Elementary hardware components work as predicted.

What is a quantum computer?

It is a computer whose elementary hardware components work according to the laws of quantum mechanics.
(The hardware components of classical digital computers work according to the laws of electronic circuits.)

There is a lot more to computation than Boolean algebra!
In going from classical to quantum computers, the concept of what is computable and what is not does not change, but the criteria of computational efficiency do.

Present Status

Laws of quantum mechanics are precisely known.
Theoretical foundation of the subject is clear.
Elementary hardware components work as predicted.
Large scale integration (say 10 or more components) is a technological challenge. Noone knows when that will arrive, or what a quantum computer will be used for.

It is inevitable

"Because the nature isn't classical, damn it . . ."

—R.P. Feynman

Laws of classical physics are convenient and useful,
and yet only approximations (that are not fully understood)
to the underlying laws of quantum physics.

Science: Observe and explain phenomena. Theorise!
Technology: Design and control phenomena. Optimise!

Yesterday's science becomes tomorrow's technology.

It is inevitable

"Because the nature isn't classical, damn it . . ."

—R.P. Feynman

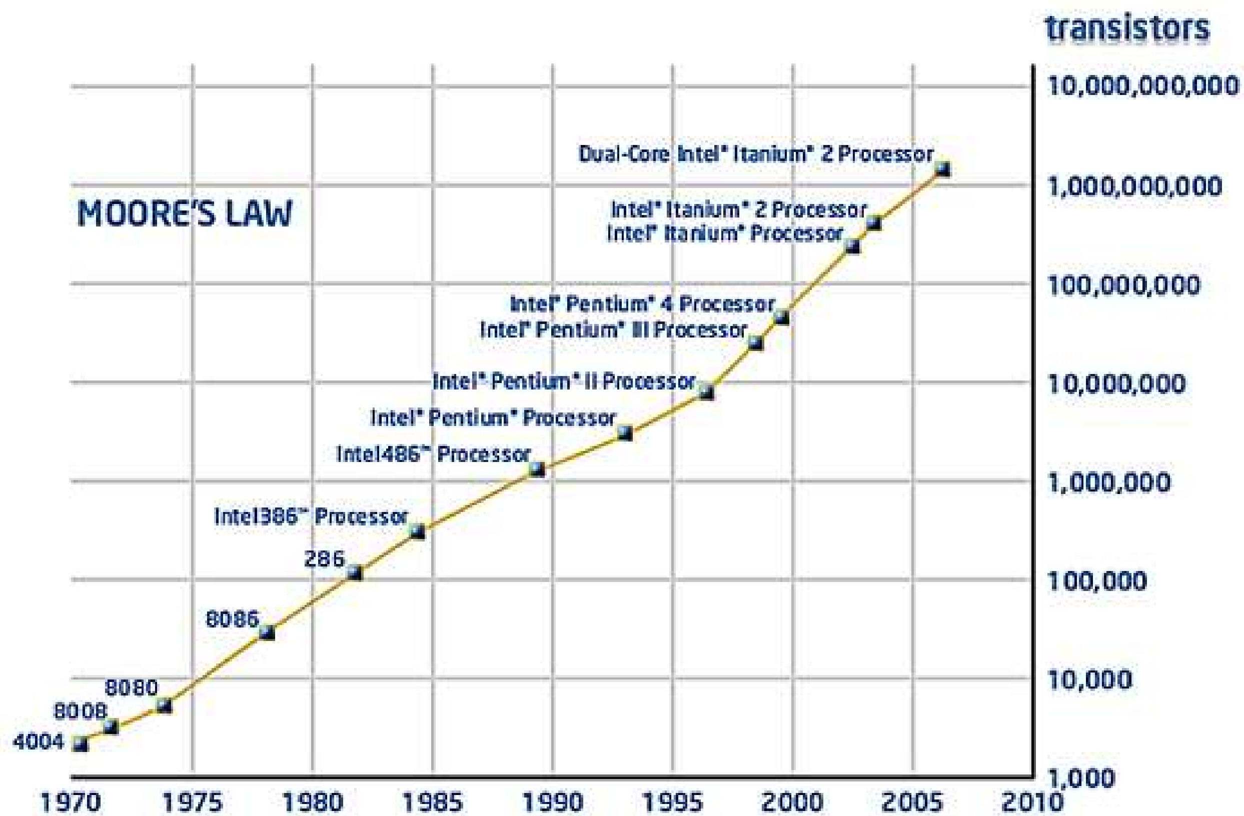
Laws of classical physics are convenient and useful, and yet only approximations (that are not fully understood) to the underlying laws of quantum physics.

Science: Observe and explain phenomena. Theorise!
Technology: Design and control phenomena. Optimise!

Yesterday's science becomes tomorrow's technology.

Quantum effects (discreteness, dispersion, tunnelling etc.) have been considered "undesirable nuisance" in the classical computer design.
Why not go to the other side, where classical effects (decoherence, thermal fluctuations etc.) become "undersirable nuisance" in the quantum computer design?

Shrinking computer circuits



Number of transistors on a chip doubles every two years.

1948: First transistor, size $\sim$ 1 cm. Today: VLSI circuits, size 45 nm.

Atomic size, 0.1 nm, is not very far!

(First nanotechnnology, and then decoherence, will have to be conquered along the way.)

It is a breakthrough

Computers are physical devices, not mere mathematical entities to implement algorithms. Quantum mechanics demonstrates that complex numbers are physical.
(We nevertheless carry the burden of history in the nomenclature—“real” and “imaginary” components.)

It is a breakthrough

Computers are physical devices, not mere mathematical entities to implement algorithms. Quantum mechanics demonstrates that complex numbers are physical. (We nevertheless carry the burden of history in the nomenclature—“real” and “imaginary” components.)

Quantum mechanics is a theory of waves. Wavefunctions can superpose, interfere, disperse and so on. Waves have been widely used in communications, but hardly any use of their properties has been made in computation.

It is a breakthrough

Computers are physical devices, not mere mathematical entities to implement algorithms. Quantum mechanics demonstrates that complex numbers are physical. (We nevertheless carry the burden of history in the nomenclature—“real” and “imaginary” components.)

Quantum mechanics is a theory of waves. Wavefunctions can superpose, interfere, disperse and so on. Waves have been widely used in communications, but hardly any use of their properties has been made in computation.

Superposition allows multiple signals at the same point at the same time. All of them can be simultaneously processed, and any one of them can be selectively observed (e.g. radio or cell-phone transmissions). This offers an SIMD parallel computing paradigm with no extra hardware. Which algorithms can exploit this?

Explorations

Discrete variables:

Qubit	Technology
Electron spin	Crystal defects
Nuclear spin	Nuclear magnetic resonance
Photon polarisation	Quantum optics, cavity QED
Two-level atom	Ion traps, Quantum dots
Magnetic flux quantum	Superconducting circuits
Non-abelian anyon	?Spin chains?

Continuous variables:

Bose-Einstein condensates, Adiabatic quantum evolution.

Explorations

Discrete variables:

Qubit	Technology
Electron spin	Crystal defects
Nuclear spin	Nuclear magnetic resonance
Photon polarisation	Quantum optics, cavity QED
Two-level atom	Ion traps, Quantum dots
Magnetic flux quantum	Superconducting circuits
Non-abelian anyon	?Spin chains?

Continuous variables:

Bose-Einstein condensates, Adiabatic quantum evolution.

Range of opinion polls on availibility of quantum computers

10 years, 20 years, 50 years, . . . , never!

TO

We already have quantum computers!

Basics

The simplest quantum system is a qubit, with two basis vectors $|0\rangle$ and $|1\rangle$ (e.g. $|\uparrow\rangle$ and $|\downarrow\rangle$ for an electron spin).
A generic qubit state is a 2-dim complex unit vector.

$$|q\rangle = \alpha|0\rangle + \beta|1\rangle, \quad |\alpha|^2 + |\beta|^2 = 1.$$

A quantum register is an ordered string of n qubits.
It is a complex unit vector in the 2^n-dim Hilbert space.

$$|x\rangle = \sum_{i_1,i_2\ldots i_n=0}^{1} c_{i_1 i_2 \ldots i_n} |x_{i_1}\rangle|x_{i_2}\rangle \ldots |x_{i_n}\rangle, \sum_{i_1,i_2\ldots i_n=0}^{1} |c_{i_1 i_2 \ldots i_n}|^2 = 1.$$

Basics

The simplest quantum system is a qubit, with two basis vectors $|0\rangle$ and $|1\rangle$ (e.g. $|\uparrow\rangle$ and $|\downarrow\rangle$ for an electron spin).
A generic qubit state is a 2-dim complex unit vector.

$$|q\rangle = \alpha|0\rangle + \beta|1\rangle, \quad |\alpha|^2 + |\beta|^2 = 1.$$

A quantum register is an ordered string of n qubits.
It is a complex unit vector in the 2^n-dim Hilbert space.

$$|x\rangle = \sum_{i_1,i_2\ldots i_n=0}^{1} c_{i_1 i_2 \ldots i_n} |x_{i_1}\rangle |x_{i_2}\rangle \ldots |x_{i_n}\rangle, \quad \sum_{i_1,i_2\ldots i_n=0}^{1} |c_{i_1 i_2 \ldots i_n}|^2 = 1.$$

A generic instruction is a rotation of the quantum state vector in the Hilbert space. It is a unitary transformation that is deterministic and fully reversible.

A measurement is a projection. In the computational basis, it yields the state $|x_{i_1}\rangle|x_{i_2}\rangle \ldots |x_{i_n}\rangle$ with probability $|c_{i_1 i_2 \ldots i_n}|^2$.
This operation is probabilistic and irreversible.

Applications

Complexity of a quantum algorithm is decided by the trade-off between the number of evolution steps and the number of states that can be coherently superposed.

The gain may be exponential, polynomial or just marginal (factor of 2).

The input and output states of a quantum computer are always mapped to classical states, through a suitable choice of basis vectors.

Applications

Complexity of a quantum algorithm is decided by the trade-off between the number of evolution steps and the number of states that can be coherently superposed.

The gain may be exponential, polynomial or just marginal (factor of 2).

The input and output states of a quantum computer are always mapped to classical states, through a suitable choice of basis vectors.

Simulation: A quantum computer can simulate quantum models efficiently. (Generically, quantum models are hard to simulate on digital computers.)

Applications

Complexity of a quantum algorithm is decided by the trade-off between the number of evolution steps and the number of states that can be coherently superposed.

The gain may be exponential, polynomial or just marginal (factor of 2).

The input and output states of a quantum computer are always mapped to classical states, through a suitable choice of basis vectors.

Simulation: A quantum computer can simulate quantum models efficiently. (Generically, quantum models are hard to simulate on digital computers.)

Cryptography: Secure key distribution protocols have been formulated and demonstrated, where an eavesdropper (unaware of the signal basis) cannot extract any information from the transmission without disturbing the signal. (The disturbance can be detected, and privacy amplification schemes allow full protection from bounded noise.)

Applications

Complexity of a quantum algorithm is decided by the trade-off between the number of evolution steps and the number of states that can be coherently superposed.

The gain may be exponential, polynomial or just marginal (factor of 2).

The input and output states of a quantum computer are always mapped to classical states, through a suitable choice of basis vectors.

Simulation: A quantum computer can simulate quantum models efficiently. (Generically, quantum models are hard to simulate on digital computers.)

Cryptography: Secure key distribution protocols have been formulated and demonstrated, where an eavesdropper (unaware of the signal basis) cannot extract any information from the transmission without disturbing the signal. (The disturbance can be detected, and privacy amplification schemes allow full protection from bounded noise.)

Pattern recognition: Clever superposition and interference can amplify the desired feature. The gain depends on the structure present in the data.

Shor's Factorisation Algorithm

Multiplication is easy, but no polynomial (in the number of digits) classical algorithm for factoring a number is known. Security of public key cryptography systems (e.g. RSA) relies on this fact.

The problem of factoring a number N can be reduced to finding the period of the function $f(x) = a^x \bmod N$. (a is chosen coprime to N, modular exponentiation is easy, number of possible remainders is limited.)

Period r: $f(0) = 1,\ f(1) = a, \ldots,\ f(r) = a^r \bmod N = 1.$

Shor's Factorisation Algorithm

Multiplication is easy, but no polynomial (in the number of digits) classical algorithm for factoring a number is known. Security of public key cryptography systems (e.g. RSA) relies on this fact.

The problem of factoring a number N can be reduced to finding the period of the function $f(x) = a^x \bmod N$. (a is chosen coprime to N, modular exponentiation is easy, number of possible remainders is limited.)
Period r: $f(0) = 1,\ f(1) = a, \ldots,\ f(r) = a^r \bmod N = 1.$

Whenever r is even, $(a^{r/2} - 1)(a^{r/2} + 1) = 0 \bmod N$. So $(a^{r/2} - 1)$ and/or $(a^{r/2} + 1)$ has a factor in common with N. (GCD is easy to calculate.)

Example: $N = 15$ and $a = 2$.

$2^x \bmod 15 = 1, 2, 4, 8, 16 \to 1, 32 \to 2, \ldots \ \Rightarrow\ r = 4, r/2 = 2.$

Both $(2^2 - 1) = 3$ and $(2^2 + 1) = 5$ are factors of 15.

Shor's Factorisation Algorithm

Multiplication is easy, but no polynomial (in the number of digits) classical algorithm for factoring a number is known. Security of public key cryptography systems (e.g. RSA) relies on this fact.

The problem of factoring a number N can be reduced to finding the period of the function $f(x) = a^x \bmod N$. (a is chosen coprime to N, modular exponentiation is easy, number of possible remainders is limited.)

Period r: $f(0) = 1,\ f(1) = a, \ldots,\ f(r) = a^r \bmod N = 1.$

Whenever r is even, $(a^{r/2} - 1)(a^{r/2} + 1) = 0 \bmod N$. So $(a^{r/2} - 1)$ and/or $(a^{r/2} + 1)$ has a factor in common with N. (GCD is easy to calculate.)

Example: $N = 15$ and $a = 2$.

$2^x \bmod 15 = 1, 2, 4, 8, 16 \rightarrow 1, 32 \rightarrow 2, \ldots \ \Rightarrow\ r = 4, r/2 = 2$.

Both $(2^2 - 1) = 3$ and $(2^2 + 1) = 5$ are factors of 15.

Periodic patterns are easily detected by Fourier Transform, which is a familiar unitary operation in quantum theory.

Quantum Fourier Transform

$$\sum_x f(x)|x\rangle = \sum_y \left(\frac{1}{\sqrt{N}} \sum_x e^{2\pi i xy/N} f(x) \right) |y\rangle$$

Let $N = 2^n$, and use the same tricks as in FFT.

In binary notation, $x = x_{n-1} \cdot 2^{n-1} + \ldots + x_1 \cdot 2 + x_0$.

$\mathrm{frac}(\frac{xy}{N}) = y_{n-1}(.x_0) + y_{n-2}(.x_1 x_0) + \ldots + y_0(.x_{n-1} \ldots x_0)$.

Quantum Fourier Transform

$$\sum_x f(x)|x\rangle = \sum_y \left(\frac{1}{\sqrt{N}} \sum_x e^{2\pi ixy/N} f(x) \right) |y\rangle$$

Let $N = 2^n$, and use the same tricks as in FFT.
In binary notation, $x = x_{n-1} \cdot 2^{n-1} + \ldots + x_1 \cdot 2 + x_0$.
$\mathrm{frac}(\frac{xy}{N}) = y_{n-1}(.x_0) + y_{n-2}(.x_1 x_0) + \ldots + y_0(.x_{n-1} \ldots x_0)$.

Unitary rotation of QFT: $|x\rangle \rightarrow \frac{1}{\sqrt{N}} \sum_y e^{2\pi ixy/N} |y\rangle$

$$= \frac{\left(|0\rangle + e^{2\pi i(.x_0)}|1\rangle\right)}{\sqrt{2}} \frac{\left(|0\rangle + e^{2\pi i(.x_1 x_0)}|1\rangle\right)}{\sqrt{2}} \cdots \frac{\left(|0\rangle + e^{2\pi i(.x_{n-1} \ldots x_0)}|1\rangle\right)}{\sqrt{2}}$$

Factorisation reduces QFT to n single qubit rotations.
The components $f(x)$ can be processed in superposition.

Quantum Fourier Transform

$$\sum_x f(x)|x\rangle = \sum_y \left(\frac{1}{\sqrt{N}} \sum_x e^{2\pi i xy/N} f(x) \right) |y\rangle$$

Let $N = 2^n$, and use the same tricks as in FFT.
In binary notation, $x = x_{n-1} \cdot 2^{n-1} + \ldots + x_1 \cdot 2 + x_0$.
$\mathrm{frac}(\frac{xy}{N}) = y_{n-1}(.x_0) + y_{n-2}(.x_1 x_0) + \ldots + y_0(.x_{n-1} \ldots x_0)$.

Unitary rotation of QFT: $|x\rangle \rightarrow \frac{1}{\sqrt{N}} \sum_y e^{2\pi i xy/N} |y\rangle$

$$= \frac{\left(|0\rangle + e^{2\pi i(.x_0)}|1\rangle\right)}{\sqrt{2}} \frac{\left(|0\rangle + e^{2\pi i(.x_1 x_0)}|1\rangle\right)}{\sqrt{2}} \ldots \frac{\left(|0\rangle + e^{2\pi i(.x_{n-1} \ldots x_0)}|1\rangle\right)}{\sqrt{2}}$$

Factorisation reduces QFT to n single qubit rotations.
The components $f(x)$ can be processed in superposition.

Fourier Transform is a multiplication by an $N \times N$ matrix.
FFT factorisation reduces the operations to $O(N \log N)$.
QFT parallelism cuts down the operations to $O((\log N)^2)$.

Quantum random walk

Efficient solutions of many practical problems require non-deterministic algorithms, which contain probabilistic branched evolution trees.

These problems are typically described using graphs, with vertices denoting the states and edges denoting the evolutionary routes.

Quantum random walk

Efficient solutions of many practical problems require non-deterministic algorithms, which contain probabilistic branched evolution trees.
These problems are typically described using graphs, with vertices denoting the states and edges denoting the evolutionary routes.

A classical computer can explore only one branch at a time, and random numbers (or equivalently coin toss instructions) are used to explore different evolutionary branches.
A particular evolution corresponds to a specific walk on the graph. The final solution is obtained by combining the results for many random walks.

Quantum random walk

Efficient solutions of many practical problems require non-deterministic algorithms, which contain probabilistic branched evolution trees.
These problems are typically described using graphs, with vertices denoting the states and edges denoting the evolutionary routes.

A classical computer can explore only one branch at a time, and random numbers (or equivalently coin toss instructions) are used to explore different evolutionary branches.
A particular evolution corresponds to a specific walk on the graph. The final solution is obtained by combining the results for many random walks.

Quantum computers can explore multiple evolutionary branches of an algorithm—in a single attempt—by using clever superpositions of states. (Coin is unnecessary.)

Quantum diffusion

Random walks represent a diffusion process.

Classical diffusion operator is the Laplacian: $\frac{\partial P}{\partial t} = \nabla^2 P$.

A spatial mode with wave vector $\vec{k}$ evolves as $\exp(-E(\vec{k})t)$, with $E(\vec{k}) \propto |\vec{k}|^2$. The slowest propagating modes (small $\vec{k}$) produce the characteristic Brownian motion signature:

$$distance \propto \sqrt{time}$$

Quantum diffusion

Random walks represent a diffusion process.
Classical diffusion operator is the Laplacian: $\frac{\partial P}{\partial t} = \nabla^2 P$.
A spatial mode with wave vector $\vec{k}$ evolves as $\exp(-E(\vec{k})t)$, with $E(\vec{k}) \propto |\vec{k}|^2$. The slowest propagating modes (small $\vec{k}$) produce the characteristic Brownian motion signature:

$$distance \propto \sqrt{time}$$

Non-relativistic quantum mechanics (Schrödinger equation) uses the same Laplacian operator, with the same scaling.
But there is an alternative. Relativistic Dirac equation uses the diffusion operator $\vec{\alpha} \cdot \vec{\nabla}$ (α_i are anticommuting objects, e.g. Pauli matrices), with $E(\vec{k}) \propto |\vec{k}|$ and the signature:

$$distance \propto time$$

Quantum diffusion

Random walks represent a diffusion process.
Classical diffusion operator is the Laplacian: $\frac{\partial P}{\partial t} = \nabla^2 P$.
A spatial mode with wave vector $\vec{k}$ evolves as $\exp(-E(\vec{k})t)$, with $E(\vec{k}) \propto |\vec{k}|^2$. The slowest propagating modes (small $\vec{k}$) produce the characteristic Brownian motion signature:

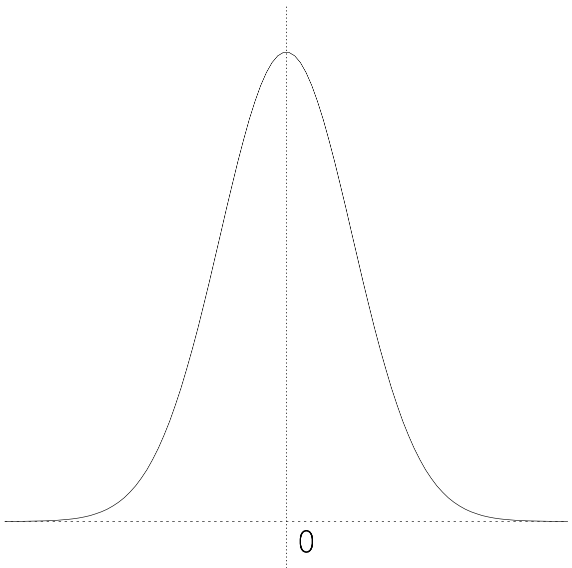
$$distance \propto \sqrt{time}$$

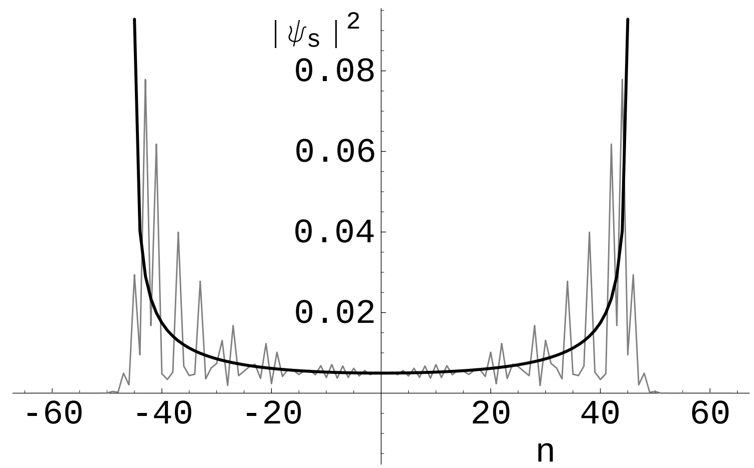
Non-relativistic quantum mechanics (Schrödinger equation) uses the same Laplacian operator, with the same scaling.
But there is an alternative. Relativistic Dirac equation uses the diffusion operator $\vec{\alpha} \cdot \vec{\nabla}$ (α_i are anticommuting objects, e.g. Pauli matrices), with $E(\vec{k}) \propto |\vec{k}|$ and the signature:

$$distance \propto time$$

Any NP-complete problem speeds up at least quadratically.

Random walk on a line





$$P(x,t) = \frac{1}{\sqrt{2\pi t}} e^{-x^2/2t}$$

$$\int P(x,t)dx = 1$$

$$\int |x| \cdot P(x,t)dx = \sqrt{\frac{2t}{\pi}}$$

$$\int x^2 P(x,t)dx = t$$

$$|\psi_s|^2 = \frac{4t^2}{\pi\sqrt{4t^2-2n^2}\ (4t^2-n^2)}$$

$$\int_{n=-\sqrt{2}t}^{\sqrt{2}t} |\psi_s|^2 dn = 1$$

$$\int_{n=-\sqrt{2}t}^{\sqrt{2}t} |n| \cdot |\psi_s|^2 dn = t$$

$$\int_{n=-\sqrt{2}t}^{\sqrt{2}t} n^2 |\psi_s|^2 dn = 2(2-\sqrt{2})t^2$$

Probability distributions for symmetric random walks:

Left: The classical one is a Gaussian.

Right: The quantum one is double peaked.

Grover's Quantum Database Search

Consider an unsorted database with N items.
Starting from an unbiased state, the desired item is to be found using the smallest number of binary oracle calls.

Grover's Quantum Database Search

Consider an unsorted database with N items.
Starting from an unbiased state, the desired item is to be found using the smallest number of binary oracle calls.

States: $|i\rangle$ any item, $|s\rangle$ starting state, $|t\rangle$ target state.

$|\langle i|s\rangle|^2 = 1/N, \quad |\langle i|t\rangle|^2 = \delta_{it}.$

Operators: Reflections along $|t\rangle$ and $|s\rangle$ directions.

$U_t = 1 - 2|t\rangle\langle t|$ (Potential energy attraction)

$U_s = 1 - 2|s\rangle\langle s|$ (Kinetic energy diffusion)

Algorithm: $(-U_s U_t)^Q |s\rangle = |t\rangle$

Solution: $(2Q+1)\sin^{-1}(1/\sqrt{N}) = \pi/2 \Longrightarrow Q = \pi\sqrt{N}/4$

Grover's Quantum Database Search

Consider an unsorted database with N items.
Starting from an unbiased state, the desired item is to be found using the smallest number of binary oracle calls.

States: $|i\rangle$ any item, $|s\rangle$ starting state, $|t\rangle$ target state.

$|\langle i|s\rangle|^2 = 1/N, \quad |\langle i|t\rangle|^2 = \delta_{it}.$

Operators: Reflections along $|t\rangle$ and $|s\rangle$ directions.

$U_t = 1 - 2|t\rangle\langle t|$ (Potential energy attraction)

$U_s = 1 - 2|s\rangle\langle s|$ (Kinetic energy diffusion)

Algorithm: $(-U_s U_t)^Q |s\rangle = |t\rangle$

Solution: $(2Q+1)\sin^{-1}(1/\sqrt{N}) = \pi/2 \Longrightarrow Q = \pi\sqrt{N}/4$

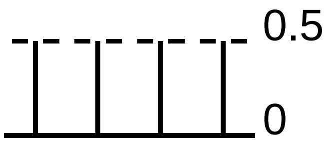
The algorithm is optimal, evolving the starting state $|s\rangle$ to the target state $|t\rangle$ along the shortest geodesic route.

Compared to its $O(\sqrt{N})$ scaling, any algorithm based on Boolean logic needs $O(N)$ oracle calls.

An example

The key feature of the algorithm is wave dynamics, and not entanglement.

Using a single oracle call, the algorithm identifies 1 out of 4 items in the database. In contrast, a Boolean algorithm identifies only 1 out of 2 items.

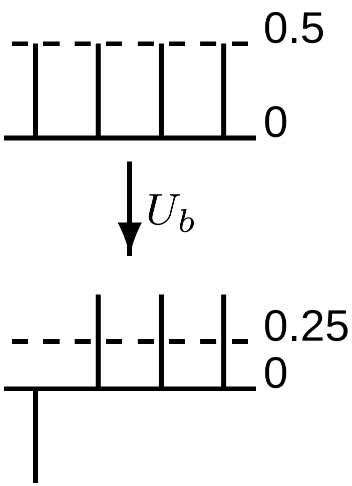
	Amplitudes	Algorithmic Steps	Physical Implementation
(1)		Uniform distribution	Equilibrium configuration

(The first item is desired by the oracle. The dashed line denotes the average amplitude.)

An example

The key feature of the algorithm is wave dynamics, and not entanglement.

Using a single oracle call, the algorithm identifies 1 out of 4 items in the database. In contrast, a Boolean algorithm identifies only 1 out of 2 items.



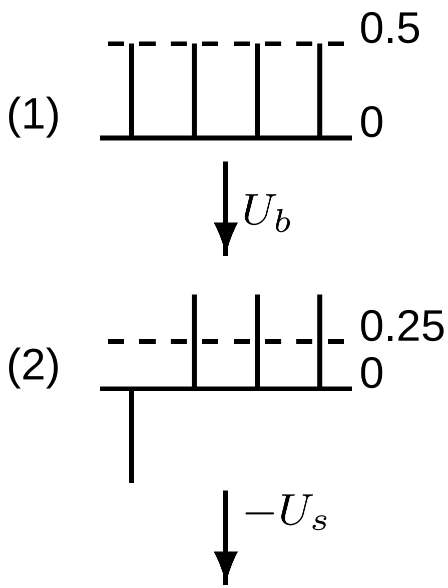
	Amplitudes	Algorithmic Steps	Physical Implementation
(1)		Uniform distribution	Equilibrium configuration
	$\downarrow U_b$	Quantum oracle	Binary question
(2)		Amplitude of desired state flipped in sign	Sudden perturbation

(The first item is desired by the oracle. The dashed line denotes the average amplitude.)

An example

The key feature of the algorithm is wave dynamics, and not entanglement.

Using a single oracle call, the algorithm identifies 1 out of 4 items in the database. In contrast, a Boolean algorithm identifies only 1 out of 2 items.

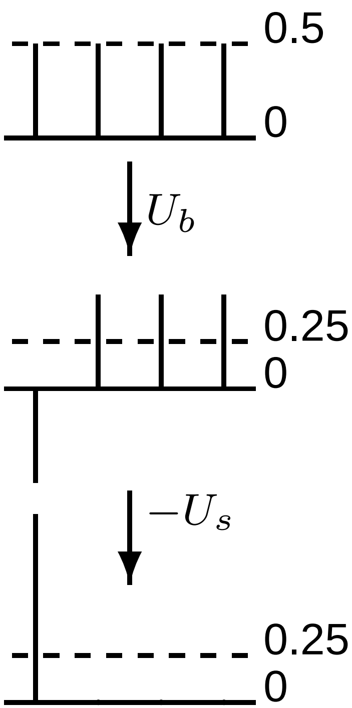


Amplitudes	Algorithmic Steps	Physical Implementation
(1)	Uniform distribution	Equilibrium configuration
U_b	Quantum oracle	Binary question
(2)	Amplitude of desired state flipped in sign	Sudden perturbation
$-U_s$	Reflection about average	Overrelaxation

(The first item is desired by the oracle. The dashed line denotes the average amplitude.)

An example

The key feature of the algorithm is wave dynamics, and not entanglement. Using a single oracle call, the algorithm identifies 1 out of 4 items in the database. In contrast, a Boolean algorithm identifies only 1 out of 2 items.



	Amplitudes	Algorithmic Steps	Physical Implementation
(1)		Uniform distribution	Equilibrium configuration
	U_b	Quantum oracle	Binary question
(2)		Amplitude of desired state flipped in sign	Sudden perturbation
	$-U_s$	Reflection about average	Overrelaxation
(3)		Desired state reached	Opposite end of oscillation
(4)	Projection	Algorithm is stopped	Measurement

(The first item is desired by the oracle. The dashed line denotes the average amplitude.)

A mechanical model

Grover's algorithm is an amplitude amplification process.
A system of coupled wave modes can execute it, provided
(1) Superposition of modes maintains phase coherence.
(2) The two reflection operations (phase changes of π for the appropriate mode) can be efficiently implemented.

A mechanical model

Grover's algorithm is an amplitude amplification process.
A system of coupled wave modes can execute it, provided
(1) Superposition of modes maintains phase coherence.
(2) The two reflection operations (phase changes of π for the appropriate mode) can be efficiently implemented.

In the quantum version, $|A|^2$ gives the probability of a state, and the algorithm solves the database search problem.
In the classical wave version, $|A|^2$ gives the energy of a mode, and the algorithm provides the fastest method for energy redistribution through the phenomenon of beats.

A mechanical model

Grover's algorithm is an amplitude amplification process.
A system of coupled wave modes can execute it, provided
(1) Superposition of modes maintains phase coherence.
(2) The two reflection operations (phase changes of π for the appropriate mode) can be efficiently implemented.

In the quantum version, $|A|^2$ gives the probability of a state, and the algorithm solves the database search problem.
In the classical wave version, $|A|^2$ gives the energy of a mode, and the algorithm provides the fastest method for energy redistribution through the phenomenon of beats.

Consider N identical coupled harmonic oscillators. Identical coupling between them is arranged by attaching them to a big oscillator through the centre-of-mass mode.
Elastic reflection of an oscillator implements the binary oracle in momentum space. Evolution by half an oscillation period implements the reflection about average operation.

Possible uses

Decoherence of quantum behaviour is extremely fast, but vibrational systems with small damping can be made easily.

Focusing of energy:

Concentration of total energy of a coupled oscillator system into a specific oscillator can have potential applications in processes that are highly sensitive to energy availability.

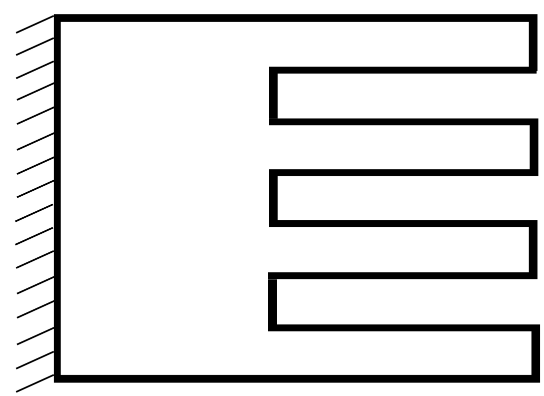
Possible uses

Decoherence of quantum behaviour is extremely fast, but vibrational systems with small damping can be made easily.

Focusing of energy:
Concentration of total energy of a coupled oscillator system into a specific oscillator can have potential applications in processes that are highly sensitive to energy availability.

Nanomechanical devices: A coupled oscillator system can provide efficient focusing of energy at a specific location, when one cannot directly control the component concerned.

For example, a comb-shaped cantilever beam can be used as a selective switch.



Catalysis: There exist many processes that need crossing of an energy threshold for completion. Their rates are typically governed by the Boltzmann factor for the energy barrier, $\exp(-E_{\rm barrier}/kT)$. Energy amplification can speed up the rates of such processes by large factors.

Catalysis: There exist many processes that need crossing of an energy threshold for completion. Their rates are typically governed by the Boltzmann factor for the energy barrier, $\exp(-E_{\rm barrier}/kT)$. Energy amplification can speed up the rates of such processes by large factors.

Dispersal of energy:

The algorithm is fully reversible, and running it backwards, i.e. $(-U_t U_s)^Q|t\rangle = |s\rangle$, distributes large initial energy in one of the oscillators equally among its partners.

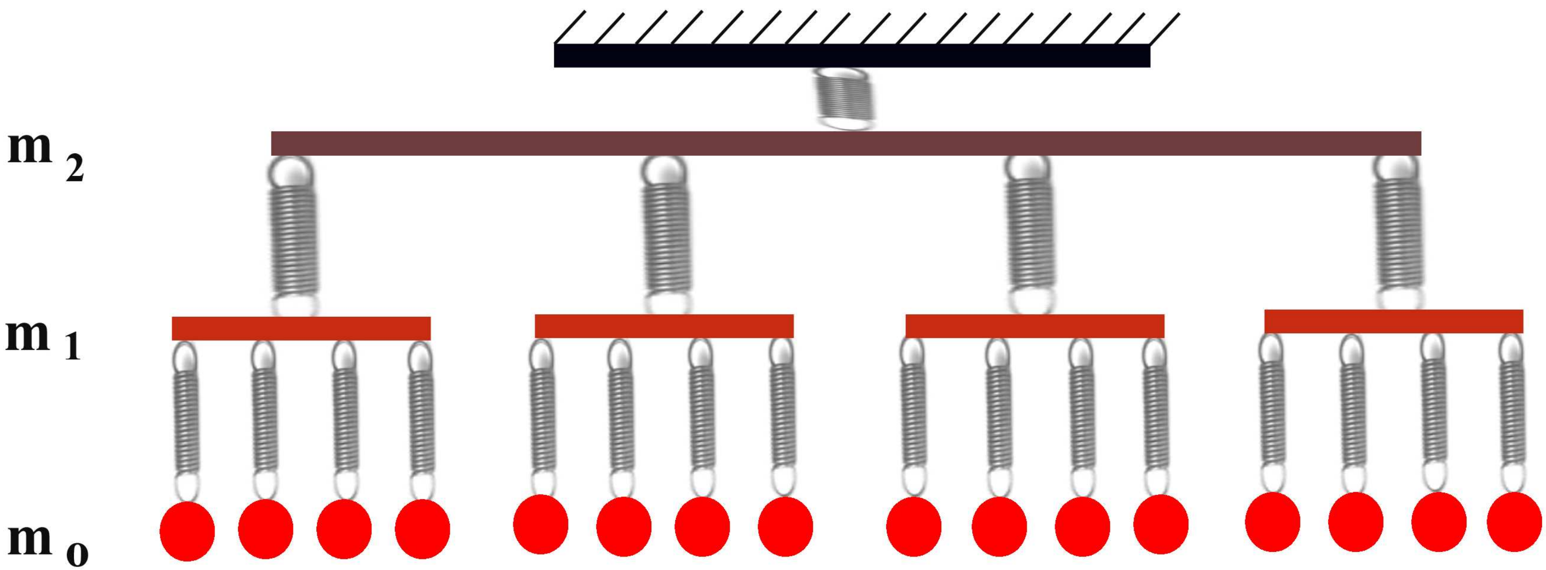
Catalysis: There exist many processes that need crossing of an energy threshold for completion. Their rates are typically governed by the Boltzmann factor for the energy barrier, $\exp(-E_{\rm barrier}/kT)$. Energy amplification can speed up the rates of such processes by large factors.

Dispersal of energy:

The algorithm is fully reversible, and running it backwards, i.e. $(-U_t U_s)^Q |t\rangle = |s\rangle$, distributes large initial energy in one of the oscillators equally among its partners.

Shock absorbers and vibrational isolation: Instead of damping a single perturbed oscillator, it is much more efficient to disperse the energy into several oscillators while damping them together.

A hierarchical system of oscillators—four small ones coupled to a big one at every level with appropriate mass, spring and damping parameters—can provide a practical realisation of this idea.



(The initial impulse is taken to be a local disturbance, which subsequently spreads out.)

Genetic languages

1. What is the information processing task carried out by the genetic code?
   Assembling molecules by picking up components from an unsorted database.
2. What is the optimal way of carrying out this task?
   Lov Grover's quantum search algorithm.
   (Requires wave dynamics.)
3. What is the signature of this algorithm?

$$(2Q+1)\sin^{-1}\frac{1}{\sqrt{N}} = \frac{\pi}{2} \implies \begin{cases} Q=1, & \text{N=4} \\ Q=2, & \text{N=10.5} \\ Q=3, & \text{N=20.2} \end{cases}$$

Lessons from Molecular Biology

Molecular biology is a nanotechnology that works—it has worked for billions of years and in an amazing variety of circumstances. Darwinian evolution has taken its basic processes to their highly optimised and essentially universal forms. By looking at them as information processing tasks, we can analyse what has been optimised and how.

Telltale signatures of quantum effects and wave dynamics show up in several instances. Examples are enzyme catalysis, photosynthesis and structure of genetic languages. Obviously, a fundamental understanding of molecular biology would have a lot to say about what we may design or convert ourselves into.

Lessons from Molecular Biology

Molecular biology is a nanotechnology that works—it has worked for billions of years and in an amazing variety of circumstances. Darwinian evolution has taken its basic processes to their highly optimised and essentially universal forms. By looking at them as information processing tasks, we can analyse what has been optimised and how.

Telltale signatures of quantum effects and wave dynamics show up in several instances. Examples are enzyme catalysis, photosynthesis and structure of genetic languages. Obviously, a fundamental understanding of molecular biology would have a lot to say about what we may design or convert ourselves into.

Enzyme Catalysis: Reaction rate enhancements range from 10^6 to 10^{12}.
Chemical industry reaches $10^3 - 10^6$.

Photosynthesis: Coherent oscillations last for longer than 500fs.
No coherence longer than 100fs was expected.

Genetic languages: No. of letters in the alphabet fit Grover's algorithm.
The languages are considered frozen accident.

References

All papers are easily accessible at http://arXiv.org/

quant-ph/9508027: P.W. Shor, *Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer*, SIAM J. Comp. 26 (1997) 1484-1509.

quant-ph/0506221: A. Patel, K.S. Raghunathan and P. Rungta, *Quantum Random Walks without Coin Toss*, Proc. "Quantum Information, Computation and Communication" (QICC 2005), pp.41-55 (Allied Publishers, New Delhi, 2006).

quant-ph/9605043: L.K. Grover, *A Fast Quantum Mechanical Algorithm for Database Search*, Proc. 28th Annual ACM Symposium on Theory of Computing (STOC'96), pp.212-219 (ACM Press, New York, 1996).

quant-ph/0609042: A. Patel, *Wave Algorithms: Optimal Database Search and Catalysis*, Proc. "Quantum Computing: BackAction 2006", pp.261-272 (AIP Conference Proceedings 864, New York, 2006).

0705.3895[q-bio.GN]: A. Patel, *Towards Understanding the Origin of Genetic Languages*, in "Quantum Aspects of Life", Eds. D. Abbott, P.C.W. Davies and A.K. Pati, (Imperial College Press, 2007).