Quantum Computing: The Final Frontier?

Colin P. Williams

Manager, Algorithms
Information Technology Program Office
Jet Propulsion Laboratory
Email: Colin.P.Williams@jpl.nasa.gov

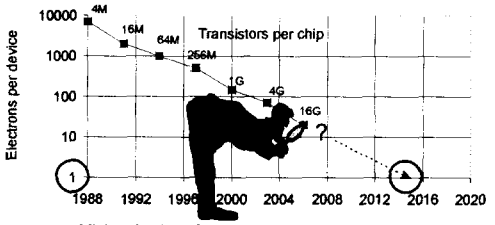
Sponsored by: ARDA

Overview

- **What is Quantum Computing?**
  - –With focus on work going on a JPL
  - –Echternach (charge), Klimeck (spin), Dowling (optics), Maleki (metrology)
- **Quantum Algorithms**
  - –Factoring composite integers
  - –Search
  - –Solving "intractable" problems (e.g., optimization, diagnosis, scheduling)
  - –Signal, image, and data processing
- **Quantum Computing Hardware**
  - –Compiling Algorithms into Hardware
  - –Superconducting and Silicon-based Quantum Computers
- **Quantum Communications**
  - –Securing Command and Control of Orbital Assets
- **Quantum Sensing**
  - –Gyroscopes
  - –Gravity gradiometers
- **Links between these areas**

Miniaturization Trend

- **Trend in miniaturization leading to quantum scales**



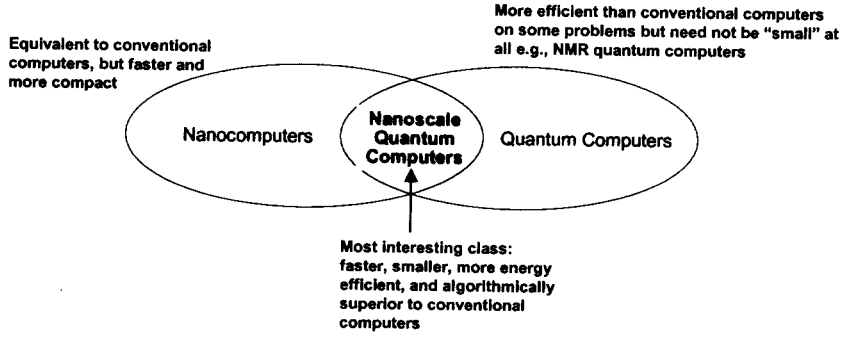
Miniaturization of computer technology as a function of time

- **Gives computers access to new repertoire of physical effects**
  - Superposition, Interference, Entanglement, Non-locality, Non-determinism, Non-clonability
  - Allows fundamentally new ***kinds*** of algorithms
- **Nanotechnology may/may not exploit all quantum phenomena**
  - To maximize impact will need to harness ***uniquely*** quantum effects, e.g., entanglement

What is Quantum Computing?

Nanotechnology vs. Quantum Technology

- **Nanocomputers compared with quantum computers**



- **Use nanofabrication techniques to assemble quantum computing hardware**

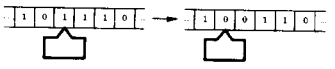
At Quantum Level Commonsense Fails

- **Theory of computation harbors implicit assumptions**
  - which cease to be true at quantum scales
- **What are these assumptions?**
  - Bit always has a value
  - This value is 0 or 1
  - Bit can be copied without error
  - Reading a bit does not change it
  - Reading a bit has no affect on other (unread) bits
- **For qubits, each assumption can fail**

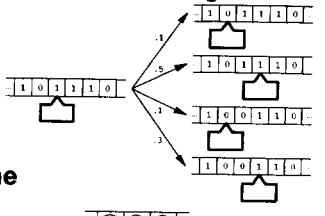
"Because nature isn't classical dammit!"
Richard Feynman

Fundamental Shift in Foundations

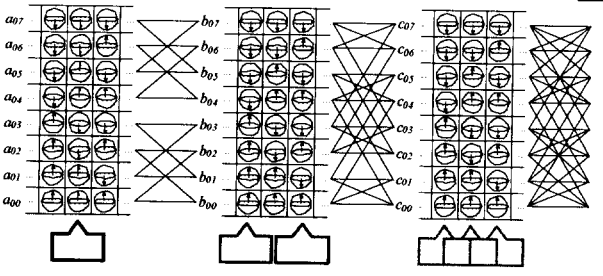
Turing Machine



Probabilistic Turing Machine



- **... becomes Quantum Turing Machine**

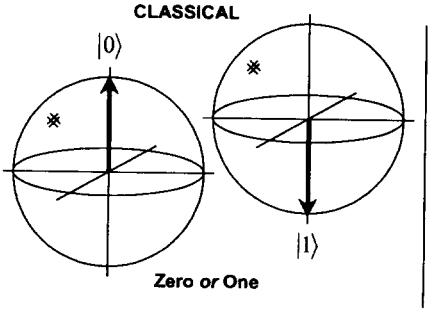


- **All computational paths pursued simultaneously**

From Bits to Qubits

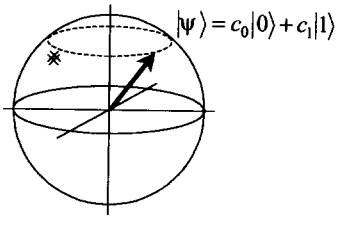
- **Use 2-state quantum systems for bits (0s and 1s) e.g. spins, polarized photons, atomic energy levels**

CLASSICAL



Zero *or* One

QUANTUM

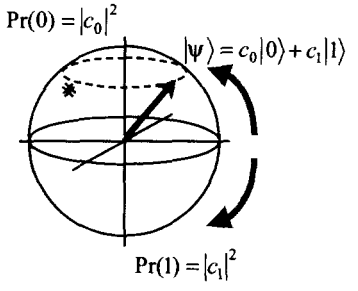


Zero *and* One

- **A qubit can exist in a *superposition* state** $|\psi\rangle = c_0|0\rangle + c_1|1\rangle$ **s.t.** $|c_0|^2 + |c_1|^2 = 1$
- **Memory register, *n* qubits** $|\psi\rangle = c_0|000\ldots0\rangle + c_1|000\ldots1\rangle + \cdots + c_{2^n-1}|111\ldots1\rangle$
- **Potential for massive parallelism ...but can't read out all answers**
  - Can sample or determine some collective property

Reading Qubit Changes Qubit

- **Physically, "readout" depends on how qubit is implemented**
  - Spin-1/2 particle: measure spin orientation
  - Polarized photon: measure plane of polarization
  - Atomic energy levels: measure energy level
- **Non-deterministic outcome**



- **Read qubit = project in $\{|0\rangle, |1\rangle\}$ basis**

Quantum Algorithms

- **Register evolves in accordance with Schrödinger eqn.**

$$i\hbar \frac{\partial|\psi\rangle}{\partial t} = H|\psi\rangle$$

- **with solution** $|\psi(t)\rangle = \exp(-iHt/\hbar)|\psi(0)\rangle = U|\psi(0)\rangle$
- **Make connection to computation:**

$$|\psi(0)\rangle \leftrightarrow \text{input data}$$
$$U \leftrightarrow \text{algorithm}$$
$$|\psi(t)\rangle \leftrightarrow \text{output before measurement}$$
$$|00\ldots0\rangle \text{ or } |00\ldots1\rangle \text{ or } \cdots \text{ or } |11\ldots1\rangle \leftrightarrow \text{output after measurement}$$

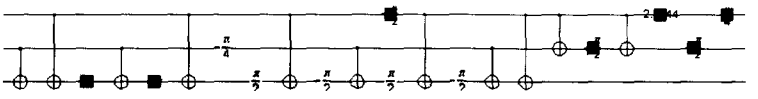
Algorithm: Specification of a sequence of unitary transformations to apply to an input quantum state, followed by a measurement

Compiling Algorithms into Quantum Circuits

- **Quantum circuit is a decomposition of desired unitary transformation into sequence of single and pairwise quantum logic gates**
- **Only requires**
  - *y*-rotations, *z*-rotations, phase-shifts, and controlled-NOT gates (CNOT)

$$R_y(\theta) = \begin{pmatrix} \cos\theta/2 & \sin\theta/2 \\ -\sin\theta/2 & \cos\theta/2 \end{pmatrix} \quad R_z(\xi) = \begin{pmatrix} e^{i\xi/2} & 0 \\ 0 & e^{-i\xi/2} \end{pmatrix} \quad Ph(\theta) = \begin{pmatrix} e^{i\theta} & 0 \\ 0 & e^{i\theta} \end{pmatrix}$$

$$CNOT = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \equiv$$



What Makes Quantum Computers So Interesting?

- **QCs take fewer steps than classical computers**
  - Not technological (faster chip) advantage
  - But complexity (fewer steps) advantage
  - Unmatchable by any classical computer
  - Potential breakthrough in solving hard computational problems
- **QCs are reversible computers**
  - Potentially energy efficient
  - Energy expended in computation is recoverable
- **QCs perform tasks that no classical computer can do**
  - Quantum teleportation
  - Utterly secure communication
  - Simulations of physical systems too complex to describe exactly

Quantum Algorithms

Quantum Algorithms

- **Exponential Speedup**
  - –Deciding whether a function is constant or balanced (Deutsch)
  - –Sampling from Fourier Transform (Simon)
  - –Factoring Integers (Shor)
  - –Simulating Quantum Systems (Abrams/Lloyd)
  - –Computing Eigenvalues (Abrams)
  - –Sampling from Wavelet Transform (Fijany / Williams)
- **Polynomial Speedup**
  - –Searching unstructured virtual databases (Grover)
  - –Solving NP-Complete/NP-Hard problems (Cerf / Grover / Williams)
  - –Finding function collisions (Brassard)
  - –Estimating Means, Medians, Maxima and Minima (Grover, Nayak/Wu, Abrams/Williams)
  - –Counting Number of Solutions (Brassard/Hoyer/Tapp)
  - –Evaluating High-dimensional Numerical Integrals (Abrams / Williams)
  - –Template Matching (Jozsa)

Quantum Algorithm I:

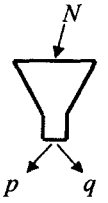
Shor's Algorithm for Factoring Composite Integers

Factoring Integers

- **Multiplication easy** $p \times q = N$
- **Factoring hard** $N \rightarrow p, q$

N = **1143816257578888676692357799761466120102182967212423625625618429...**

...357069352457338978305971235639587050589890751475992900268795435 41

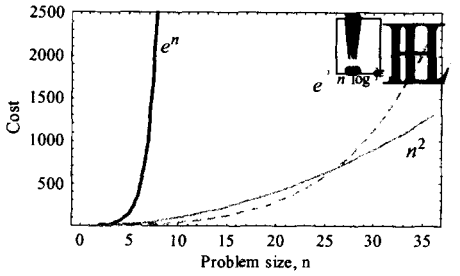


p = 327691329932667095499619881908344614131776429679929425397982885 33

q = **3490529510847650949147849619903898133417764638493387843990820577**

Complexity of Factoring Integers

- **Number Field Sieve** $O(e^{n^{1/3}(\log n)^{2/3}})$ **sub-exponential (hard!)**



- **Why does anyone care?**
- **Security of widely used public key cryptosystems rests on the presumption that factoring is hard, e.g., RSA**

RSA Public Key Cryptosystem

Create Keys	**1. Find two primes, and compute their product $N = p\,q$** **2. Find integer d coprime to $(p\text{-}1)(q\text{-}1)$** **3. Compute e from $e\,d = 1 \bmod (p\text{-}1)(q\text{-}1)$** **4. Broadcast public key (e,N), keep private key (d,N) secret**
Encrypt	**5. Represent message P as a sequence of integers $\{M_i\}$** **6. Encrypt M_i using public key and rule $E_i = M_i^e \bmod N$**
Decrypt	**7. Decrypt using private key and rule $M_i = E_i^d \bmod N$** **8. Reconvert the $\{M_i\}$ back to the plaintext P**

- **As public key (e, N) known, can crack RSA if you can factor N into $N = p\,q$ by computing private key (d, N) from $e\,d = 1 \bmod (p\text{-}1)(q\text{-}1)$**

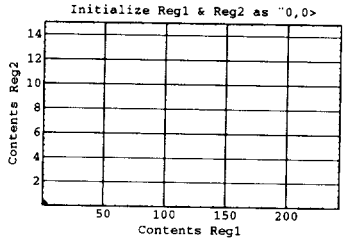
Example of RSA

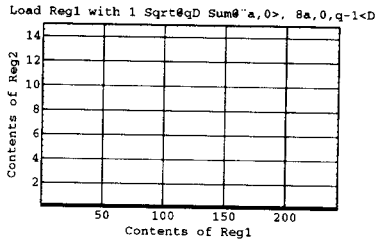
```
In[29]:= [illegible]PublicKey, $PrivateKey = CreatePublicKeyAndPrivateKey[[illegible]]

      Picking p: p = 3097172369
      Picking q: q = 3782480549
      Hence n = p q = 11714994242642750581
      Picking large integer d, co-prime to n: d = 7520374751991265811
      Computing modular inverse, e, from e d = 1 mod [illegible] 9871244581433966043
      Public  Key is [illegible] n = [illegible]871244581433966043, 11714994242642750581
      Private Key is [illegible] n = [illegible]520374751991265811, 11714994242642750581

In[30]:= cipherText = EncryptRSA[[illegible]ngry. Let's eat! ", $PublicKey[illegible]

Out[30]= [illegible]37662632885750605, 4223282963866241971, 8515734954729530610,
       572105026579800127, 3125477641371647366, 8785778425474049423, 116095988027245517,
       1843196734898219 67, 4095890900271762030, 5711708545539327862, 5188837378111696662

In[31]:= DecryptRSA[[illegible]ext, $PrivateKey[illegible]

Out[31]= I'm hungry. Let's eat!
```

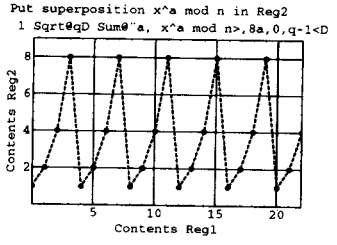

Factoring via Period Finding

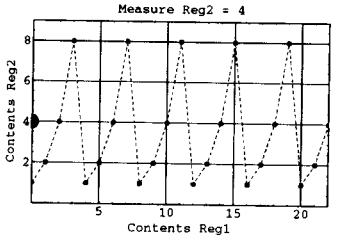
- **Can factor integers by finding period of a function related to the factors**
- **Classical (*inefficient*) algorithm**
- **Example: factor $N = 15$**
  - **Choose random integer x that is coprime to N**
    - e.g. $x = 2$ will suffice because gcd(2, 15) = 1
  - **Compute the sequence of integers x^i mod N, giving:**
    - 2^0 mod 15, 2^1 mod 15, … = $\boxed{1, 2, 4, 8}$ 1, 2, 4, 8, 1, 2, 4, 8 …
  - **Sequence is periodic, with period $r = 4$**
  - **Factors of N given by gcd($x^{r/2} \pm 1$, N)**
  - **Gives 15 = $p\,q$ where p = gcd(5,15) = 5, q = gcd(3,15) = 3**
- **But there is a fast quantum algorithm for period finding**
  - **Based on sampling from Fourier transform of this periodic sequence**

Quantum Factoring I: Periodic State

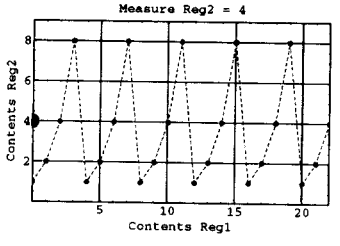


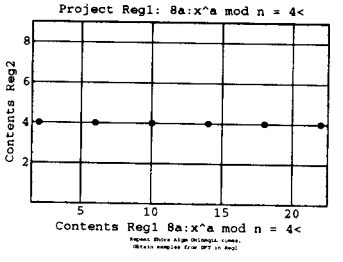


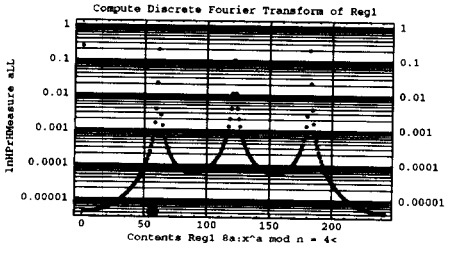


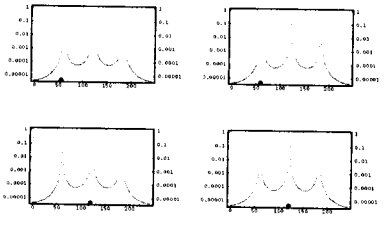


Quantum Factoring II: Find Period







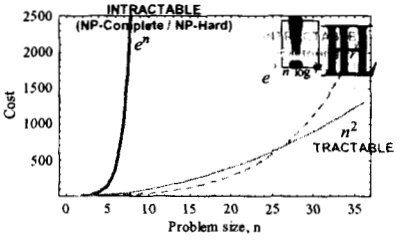


Quantum Algorithm II:

Using Quantum Search to Speed up Solution of NP-Complete Problems

NASA-Relevant Computations

- **Autonomy relies on solving NP-Complete/NP-Hard problems**
  - –Diagnosis
  - –Planning
  - –Scheduling
  - –Combinatorial Optimization
  - –Learning
  - –Constraint Satisfaction
  - –etc ...
- **Image Interpretation**
  - –Change detection
  - –Superresolution
  - –Pattern recognition



Solving one type of NP-Hard problem efficiently would solve ALL types of NP-Hard problems efficiently as you can easily interconvert them

- **Can't tame NP-Hard problems with conventional computers**
- **But quantum computers can speed up computations by:**
  - –Exponential factor,
  - –Polynomial factor, or
  - –Not at all
  - –So possibility exists for fundamental ***algorithmic*** advance

Quantum Search Algorithm

- **Invented by Lov Grover, Bell Labs, in 1996**
  - L. Grover, "*A Fast Quantum Mechanical Algorithm for Database Search*", in Proceedings of the 28th Annual ACM Symposium on the Theory of Computing (1996) pp212-219.
  - G. Brassard, "*Searching a Quantum Phone Book*", Science, January 31st (1997) pp.627-628.
- **Problem: Find the name of the person in a telephone directory who has a prescribed telephone number**
  - Suppose N entries in directory
  - Classical: need $O(N)$ queries in worst case
  - Quantum: need $O(N^{1/2})$ queries in worst case
- **Gives *polynomial* speedup**
- **Use as subroutine in higher-level quantum algorithms**

How Quantum Search Works

$$f_t(x) = \begin{cases} 0 & x \neq t \\ 1 & x = t \end{cases}$$

- **Knowledge of database encoded in an "oracle" function**
  - x is the index of an item in the database
  - Target entry has index $x = t$
  - Oracle returns $f_t(t) = 1$, $f_t(x) = 0$ otherwise
- **Use "oracle" to build an "amplitude amplification operator", Q**

$$\hat{Q} = -\hat{U} \cdot \hat{I}_s \cdot \hat{U}^{-1} \cdot \hat{I}_{f_t}$$

  - where $|s\rangle$ is a superposition of equally weighted indices
  - $|t\rangle$ is the (unknown) target index that you are seeking
  - $\hat{I}_s = 1 - 2|s\rangle\langle s|$ is a unitary operator
  - $\hat{I}_{f_t} = 1 - 2|t\rangle\langle t|$ is the unitary operator representing the oracle
  - $\hat{U}$ is any unitary matrix having only non-zero elements

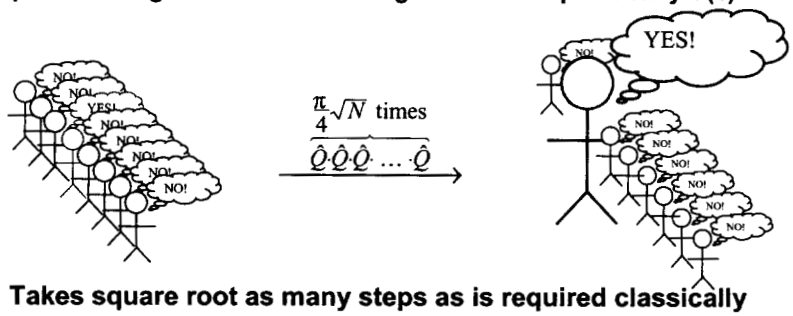
Amplitude Amplification Boosts "Signal"

Step 1: Create equally weighted superposition of all *N* candidates

Step 2: Synthesize amplitude amplification op.

Step 3: Apply Q $\frac{\pi}{4}\sqrt{N}$ times

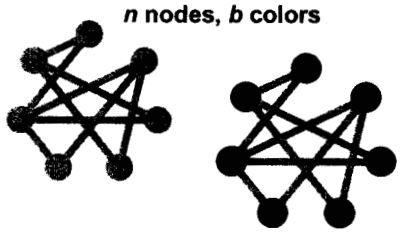
Step 4: Read register – will obtain target index with probability *O*(1)



- **Takes square root as many steps as is required classically**
- **Fundamental algorithmic advance that is only possible on a quantum computer**

What about the NP-Hard Problems?

***n* nodes, *b* colors**



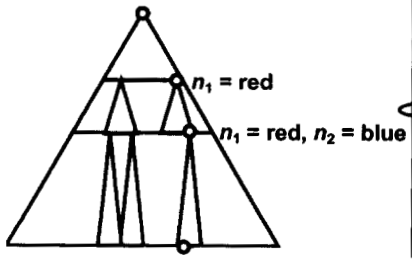
Nested Quantum Search

Step 1: Superposition of consistent partial solutions at intermediate level

Step 2: Perform amplitude amplification in the subspace of their descendants

Step 3: Nest Step 1 inside Step 2

Induces tree-structured search space



Comparison

- **Best classical tree search O($b^{0.446n}$)**
- **Naïve Quantum Search O($b^{0.5n}$)**
- **Structured Quantum Search O($b^{0.333n}$)**

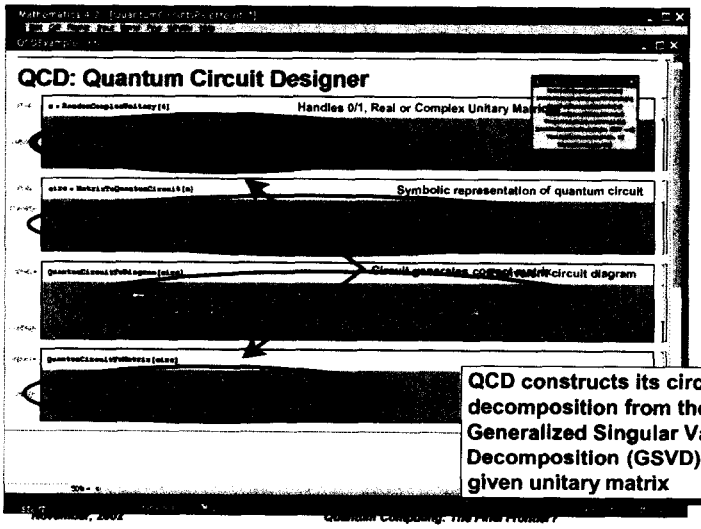
- N. Cerf, L. Grover, C. P. Williams, "*Nested Quantum Search and Structured Problems*," Phys. Rev. A, 61, 032303, 9th February (2000)
- C. P. Williams, "*Quantum Search Algorithms in Science and Engineering*", Colin P. Williams, Computing in Science and Engineering, IEEE Computer Society, April (2001).

Making the Connection to Quantum Computer Hardware

Compiling Quantum Algorithms into Quantum Circuits

QCD: Quantum Circuit Design Tool

- **QCD: *Mathematica*-based circuit design tool**



QCD constructs its circuit decomposition from the Generalized Singular Value Decomposition (GSVD) of the given unitary matrix

Generalized Singular Value Decomposition

- **GSVD exploits fact that blocks of a partitioned unitary matrix have highly related singular value decompositions** *(see Golub & van Loan, "Matrix Computations", p.77)*
- **GSVD decomposition of a $2^n \times 2^n$ unitary matrix**

$$\boxed{U = \underbrace{\left(\begin{array}{c|c} U_{00} & U_{01} \\ \hline U_{10} & U_{11} \end{array}\right)}_{2^n} = \left(\begin{array}{c|c} L_0 & 0 \\ \hline 0 & L_1 \end{array}\right) \cdot \left(\begin{array}{c|c} D_{00} & D_{01} \\ \hline D_{10} & D_{11} \end{array}\right) \cdot \left(\begin{array}{c|c} R_0 & 0 \\ \hline 0 & R_1 \end{array}\right)}$$

- L_0, L_1, R_0, R_1, **are $2^{n-1} \times 2^{n-1}$ unitary matrices**
- $D_{00} = D_{11} = diag(C_1, C_2, \ldots, C_{2^{n-1}})$
- $D_{10} = -D_{01} = diag(S_1, S_2, \ldots, S_{2^{n-1}})$

Apply Recursively ...

- **Recurse until factors are direct sums of 1-qubit gates**

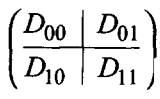
$$\left(\begin{array}{c|c} L_0 & 0 \\ \hline 0 & L_1 \end{array}\right) = \left(\begin{array}{c|c} \left(\begin{array}{c|c} L_0' & 0 \\ \hline 0 & L_1' \end{array}\right) \cdot \left(\begin{array}{c|c} D_{00}' & D_{01}' \\ \hline D_{10}' & D_{11}' \end{array}\right) \cdot \left(\begin{array}{c|c} R_0' & 0 \\ \hline 0 & R_1' \end{array}\right) & 0 \\ \hline 0 & \left(\begin{array}{c|c} L_0'' & 0 \\ \hline 0 & L_1'' \end{array}\right) \cdot \left(\begin{array}{c|c} D_{00}'' & D_{01}'' \\ \hline D_{10}'' & D_{11}'' \end{array}\right) \cdot \left(\begin{array}{c|c} R_0'' & 0 \\ \hline 0 & R_1'' \end{array}\right) \end{array}\right)$$

$$= \left(\begin{array}{c|c|c} L_0' & & 0 \\ \hline & L_1' & \\ \hline 0 & & L_0'' \\ \hline & & & L_1'' \end{array}\right) \cdot \underbrace{\left(\begin{array}{c|c|c} D_{00}' & D_{01}' & 0 \\ \hline D_{10}' & D_{11}' & \\ \hline 0 & D_{00}'' & D_{01}'' \\ \hline & D_{10}'' & D_{11}'' \end{array}\right)} \cdot \left(\begin{array}{c|c|c} R_0' & & 0 \\ \hline & R_1' & \\ \hline 0 & R_0'' & \\ \hline & & R_1'' \end{array}\right)$$

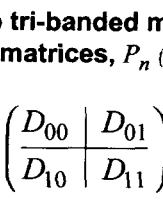
1-qubit gates **1-qubit gates**

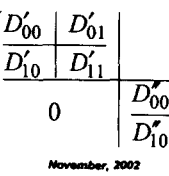
Central matrix = blocks of tri-diagonal matrices. Needs special handling

Tri-banded to Block-diagonal

- **Central matrix** $\left(\begin{array}{c|c} D_{00} & D_{01} \\ \hline D_{10} & D_{11} \end{array}\right) \equiv$  **is always tri-banded**

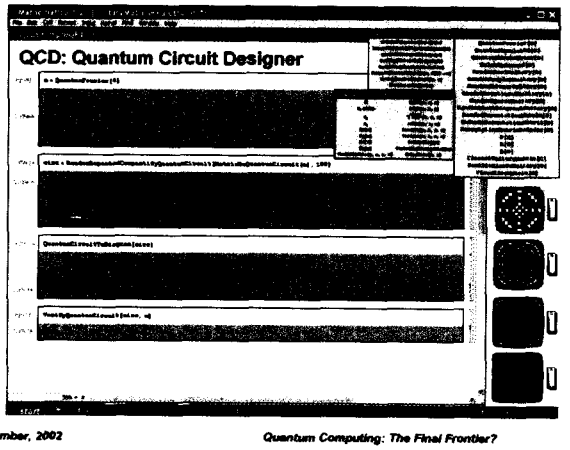
- **Can map tri-banded matrix to block-diagonal matrix using qubit reversal matrices,** P_n *(cascaded SWAP gates)*

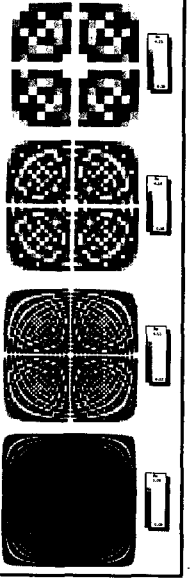
$$\left(\begin{array}{c|c} D_{00} & D_{01} \\ \hline D_{10} & D_{11} \end{array}\right) \equiv$$  $$= P_n^{-1} \cdot \left(\begin{array}{c|c} \blacksquare & 0 \\ \hline 0 & \blacksquare \end{array}\right) \cdot P_n$$

$$\left(\begin{array}{c|c|c|c} D'_{00} & D'_{01} & \multicolumn{2}{c}{0} \\ \hline D'_{10} & D'_{11} & & \\ \hline \multicolumn{2}{c|}{0} & D''_{00} & D''_{01} \\ \hline & & D''_{10} & D''_{11} \end{array}\right) \equiv$$  $$= (P_{n-1}^{-1} \oplus P_{n-1}^{-1}) \cdot \left(\begin{array}{c|c|c|c} \blacksquare & 0 & 0 & 0 \\ \hline 0 & \blacksquare & 0 & 0 \\ \hline 0 & 0 & \blacksquare & 0 \\ \hline 0 & 0 & 0 & \blacksquare \end{array}\right) \cdot (P_{n-1} \oplus P_{n-1})$$

Quantum Fourier Transform

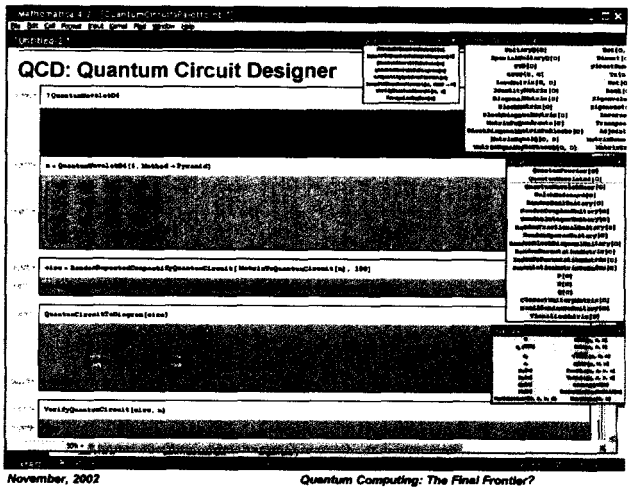
- **QCD can detect special structure if it exists**
  - E.g. QCD finds a compact circuit for QFT
  - Comparable to direct conversion of usual QFT circuit which involves conditional gates

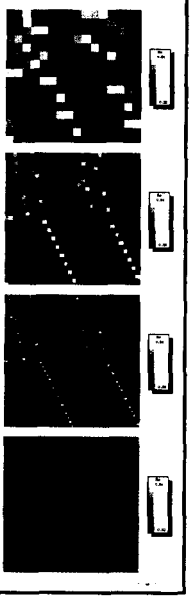




Quantum Wavelet (D4) Transform

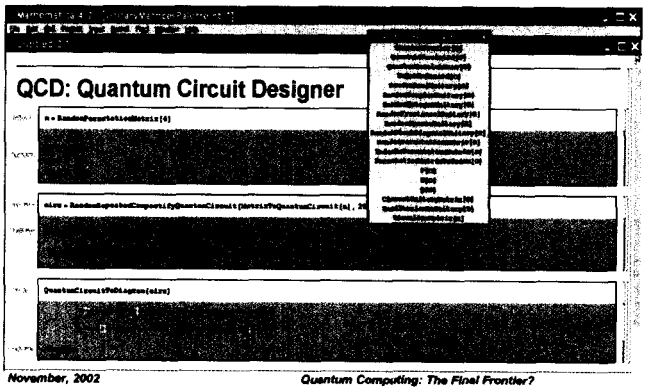
- **QWT (in pyramid algorithm) also has special structure**
  - QCD also finds compact circuits for QWT





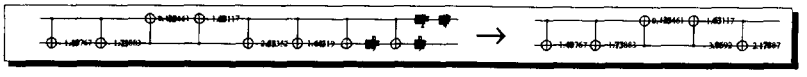
Permutations

- **Efficiency of many algorithms related to the efficiency of performing permutations of the amplitudes**
- **Permutation matrices are 0/1 matrices**
  - Can decompose into Fredkin/Toffoli gates
- **But more efficient to use 1-qubit rotations**



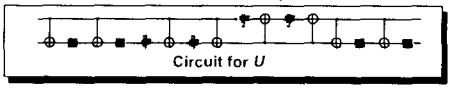
Deterministic Compactification

- **QCD attempts to find the smallest circuit for a given *U* by applying circuit compactification rules**
  - Eliminate gates from the circuit while preserving correctness, e.g.,

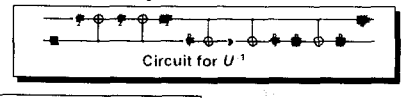


- **QCD considers depth of circuits for *U* and U^{-1}**
  - If CircuitDepth(U) < CircuitDepth(U^{-1}) use circuit for U
  - If CircuitDepth(U^{-1}) < CircuitDepth(U) use inverse circuit for U^{-1}, e.g.,

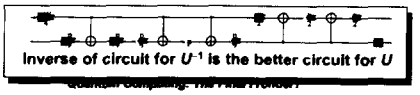
$$U=\begin{pmatrix}1&0&0&0\\0&0&0&1\\0&1&0&0\\0&0&1&0\end{pmatrix}$$



Circuit for *U*



Circuit for U^{-1}



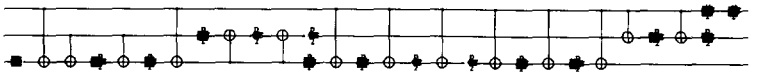
Inverse of circuit for U^{-1} is the better circuit for *U*

Randomized Compactification

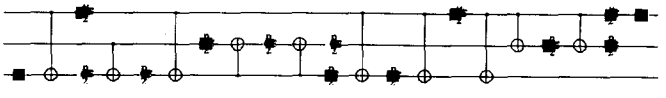
- **QCD selects a sub-circuit, computes implied unitary matrix, redesigns a circuit for it, and accepts the result if of lower depth**
  - Compactifies across boundaries of adjacent conditional gates, e.g.,

Target matrix =
$$\begin{pmatrix}1&0&0&0&0&0&0&0\\0&0&0&1&0&0&0&0\\0&0&1&0&0&0&0&0\\0&1&0&0&0&0&0&0\\0&0&0&0&0&0&0&1\\0&0&0&0&1&0&0&0\\0&0&0&0&0&1&0&0\\0&0&0&0&0&0&1&0\end{pmatrix}$$

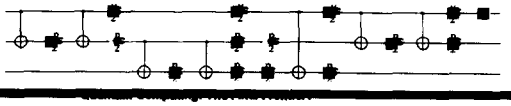
Raw output =



Deterministic Compactification =



Randomized Compactification =



Using QCD in Higher-level Quantum Algorithms

- Synthesizing Pure States
- Synthesizing Arbitrary Mixed States
- Quantum Simulation and Data Processing

Application: Pure State Synthesis

- **How do we synthesize an arbitrary *n*-qubit pure state** $|\psi\rangle = \sum_{i=0}^{2^n-1} c_i |i\rangle \equiv \begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_{2^n-1} \end{pmatrix}$ **?**

`Algorithm SynthesizePureState:`

Step 1: w.l.o.g. assume amplitude $c_0 \neq 0$ (otherwise permute basis until $c_0 \neq 0$)

Step 2: Construct the matrix M, defined by

$$M = \left(\begin{array}{c|ccc} c_0 & & & \\ \hline c_1 & 1 & & \\ \vdots & & 1 & \\ \vdots & & & 1 \\ c_{2^n-1} & & & & 1 \end{array}\right)$$

Step 3: Use Gram-Schmidt process to fix first column as $|\psi\rangle$ and compute orthonormal columns for the rest of the matrix

Step 4: Resulting matrix is unitary. Use QCD to compute a circuit for it.

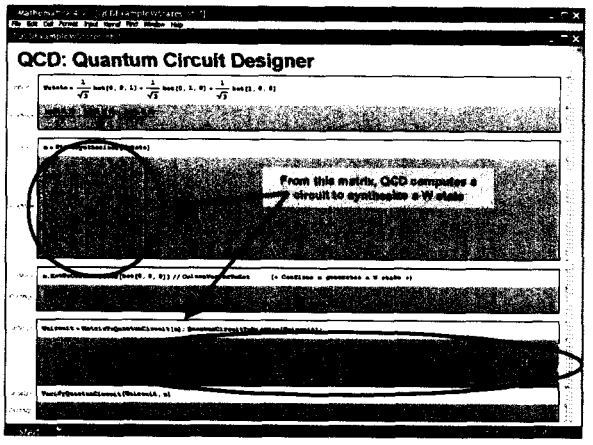
Output: A circuit for synthesizing an arbitrary pure state $|\psi\rangle$

Example: Synthesizing *W* States

- **Goal: to make *W* state** $|W\rangle = \frac{1}{\sqrt{3}}\left(|001\rangle + |010\rangle + |100\rangle\right)$ **starting from** $|000\rangle$
  - Inequivalent to GHZ

$|W\rangle$ complete the matrix with orthonormal vectors

$$U = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ \frac{1}{\sqrt{3}} & -\frac{1}{\sqrt{6}} & 0 & -\frac{1}{\sqrt{2}} & 0 & 0 & 0 & 0 \\ \frac{1}{\sqrt{3}} & \sqrt{\frac{2}{3}} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{\sqrt{3}} & -\frac{1}{\sqrt{6}} & 0 & \frac{1}{\sqrt{2}} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$



Application: Mixed State Synthesis

- **How do we synthesize an arbitrary mixed state, ρ ?**

Algorithm SynthesizeMixedState:

Step 1: Compute the spectral decomposition of $\rho = \sum p_i |i\rangle\langle i|$

Step 2: Compute the unitary operators U_i s.t. $|i\rangle = U_i|0\rangle$

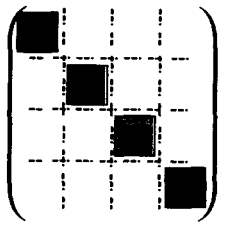
Step 3: Compute a circuit for performing $U = U_1 \oplus U_2 \oplus \ldots$

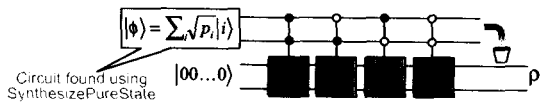
Step 4: Compute a circuit for preparing the "loaded dice" state $|\phi\rangle = \sum_i \sqrt{p_i}|i\rangle$

Step 5: Compute the input state $|\psi\rangle = |\phi\rangle \otimes |00\ldots0\rangle$

Step 6: Push this state through U, and trace over the control qubits, C, i.e. perform the computation $\mathrm{Tr}_C(U|\psi\rangle\langle\psi|U^+) = \rho$

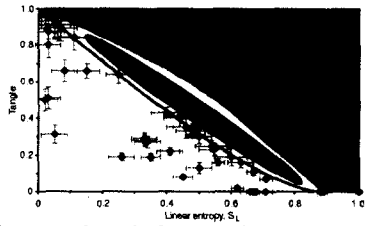
Output: A circuit for sythesizing an arbitrary mixed state ρ





Example: Synthesizing Maximal States

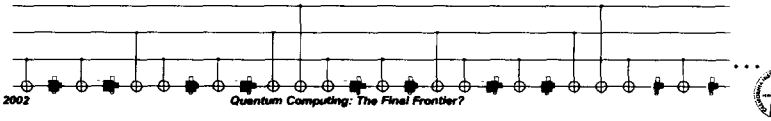
- **Maximal states lie on a boundary in the tangle/entropy plane separating physically allowed states (white) from physically impossible ones (gray)**
- **Maximal states have a greater tangle than Werner states (black curve)**



$$\rho_{maximal} = \begin{pmatrix} \frac{1}{3} & 0 & 0 & \frac{1}{3} \\ 0 & \frac{1}{3} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \frac{1}{3} & 0 & 0 & \frac{1}{3} \end{pmatrix}$$

- **Not previously known how to generate maximal states**
  - See White, James, Munro, & Kwiat "Exploring Hilbert Space" PRA 01/2001
- **QCD computes circuits to generate such states automatically**

```
In[]:= QuantumCircuitToDiagram[ SynthesizeMixedState[ρmaximal] ]
Out[]=
```



Towards Quantum Signal, Image and Data Processing

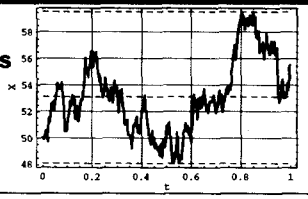
Signal, Data and Image Processing

- **Earth Sciences and Space Sciences Enterprises**
- **Signal, image and data processing fundamentally different on a quantum computer than classical computer**
  - –Classical-to-quantum data encoding
    - •Linear cost
  - –Quantum processing
    - •Some operations yield exponential speedups
    - •e.g., quantum versions of Fourier, wavelet, and cosine transforms
  - –Quantum-to-classical readout
    - •Cannot "see" result in conventional sense
    - •Can sample from, or obtain collective properties of, processed signal, image or data
- **Can process an image exponentially more efficiently, report on a property of interest, but be unable to display the result**
  - –Quantum world strongly distinguishes truth from proof
- **Let's look at how to enter data into a quantum computer**

Data-Entry on a Quantum Computer

- **Encode 2^n data values as the amplitudes of just n qubits**

$$|\psi\rangle = \sum_{i=0}^{2^n-1} c_i |i\rangle \equiv \begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_{2^n-1} \end{pmatrix}$$



Algorithm DataEntry:

Step 1: Normalize the data vector, and pad it to length $2^{\lceil \log_2 \|c\| \rceil}$, i.e., compute $c_i' \leftarrow \frac{c_i}{\sum_i |c_i|^2}$

Step 2: Interpret c_i' as the amplitudes of the pure state $|\psi'\rangle$

Step 3: w.l.o.g. assume amplitude $c_0' \neq 0$ (otherwise permute basis until $c_0' \neq 0$)

Step 4: Construct the matrix M defined by:

$$M = \left(\begin{array}{c|cccc} c_0' & & & & \\ \hline c_1' & 1 & & & \\ \vdots & & 1 & & \\ \vdots & & & 1 & \\ c_{2^n-1}' & & & & 1 \end{array}\right)$$

Step 5: Use Gram-Schmidt process to fix first column as $|\psi'\rangle$ and compute orthonormal columns for the rest of the matrix

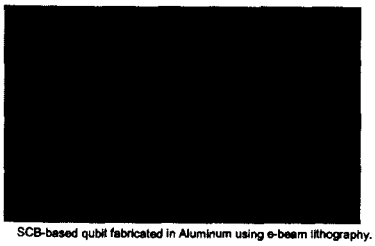
Step 6: Map this unitary matrix into an equivalent quantum circuit using QCD circuit design tool

Output: A circuit for synthesizing an arbitrary data input to a quantum computer

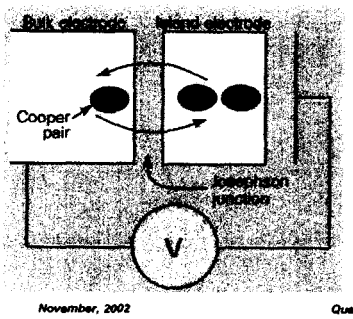
Using QCD in Quantum Hardware Design

Charge-based Qubits

- Collaboration with Pierre Echternach (JPL)
- **Qubits as Single Cooper Pair Boxes**
  - Cooper pairs tunnel through Josephson junction onto island
  - Qubit encoded as the number of Cooper pairs on the island
  - Coherent oscillations in the number of pairs observed in 1999



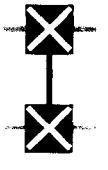
SCB-based qubit fabricated in Aluminum using e-beam lithography.



- **How do we make *CNOT*?**
  - From an easier-to-achieve gate - iSWAP

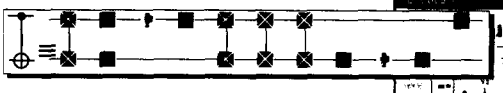
$$iSWAP = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & i & 0 \\ 0 & i & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

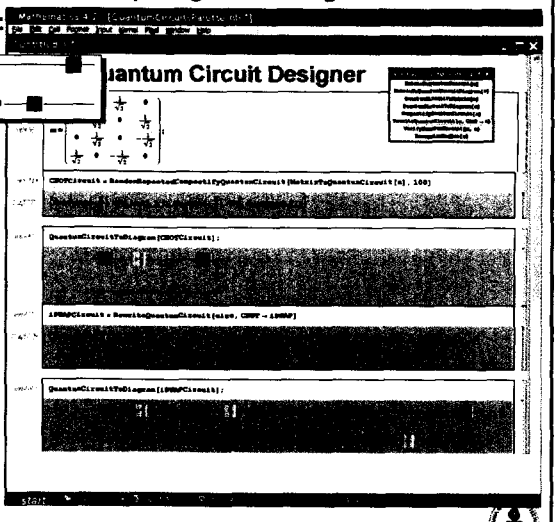
iSWAP circuit icon



Tailoring Circuits to Charge-based Qubits

- **Better to use *iSWAP* than *CNOT* as the 2-qubit gate in charge-based QCs**
- **Can achieve *CNOT* as follows:**

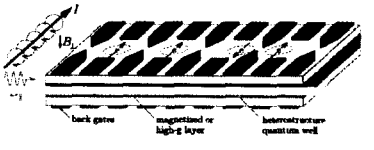




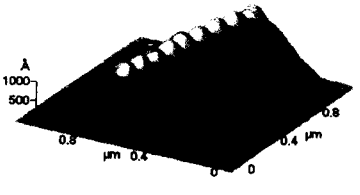
- **So {all 1-qubit gates} ∪ { *iSWAP* } is universal set**
- **Example shows a charge-based circuit for entangling charge-number states**
- **N.B.** $iSWAP^{-1} = iSWAP^{3}$

Spin-based Qubits

- Collaboration with Gerhard Klimeck (JPL)
- **Qubits as spin-based quantum dots**
  - ***e.g., Loss and Di Vincenzo, "Quantum Computation with Quantum Dots", Phys. Rev. A, 57, 1 (1998)***

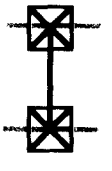


- **How do we make *CNOT*?**
  - From an easier-to-achieve gate - $\sqrt{SWAP}$



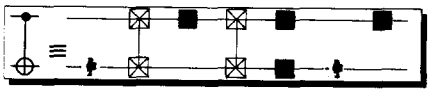
$$\sqrt{SWAP} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1+i}{2} & \frac{1-i}{2} & 0 \\ 0 & \frac{1-i}{2} & \frac{1+i}{2} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$\sqrt{SWAP}$ **circuit icon**

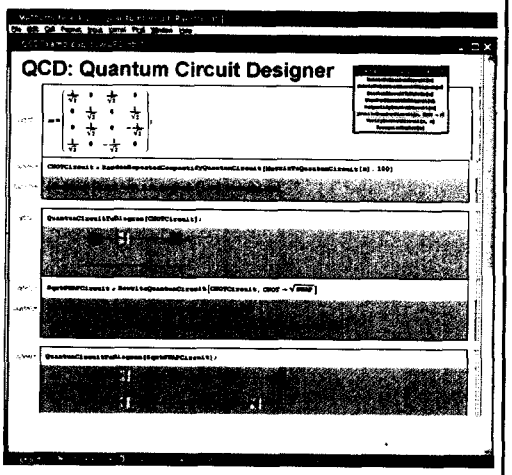


Tailoring Circuits to Spin-based Qubits

- Better to use $\sqrt{SWAP}$ than *CNOT* as the 2-qubit gate in spintronic QCs
- Can achieve *CNOT* as follows:

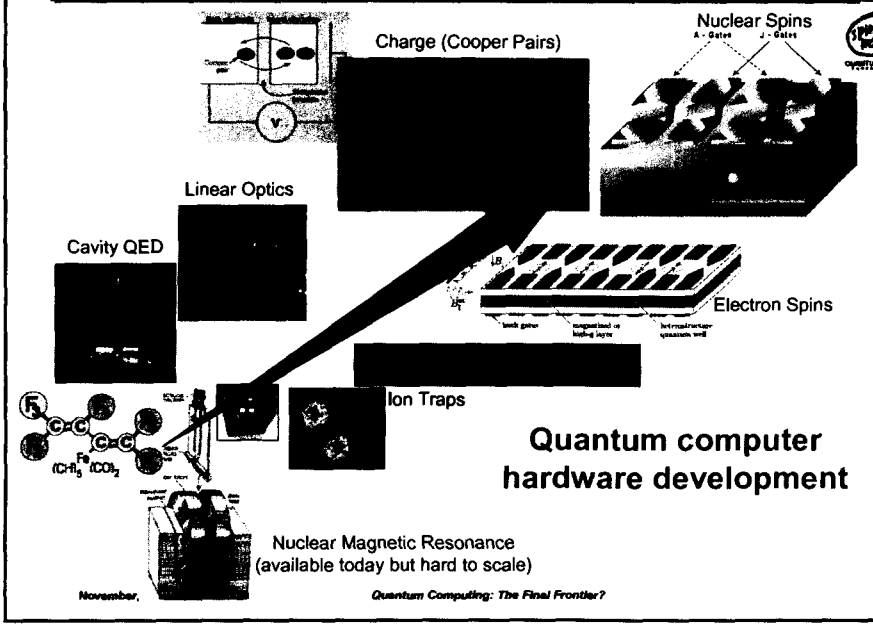


- So {all 1-qubit gates} $\cup$ {$\sqrt{SWAP}$ } is universal set
- Example shows a spintronic circuit for entangling spins
- Already started extension to decoherence-free basis in which 1 logical qubit is encoded in 3 physical spins



Quantum Computing Hardware

Progress in Quantum Computing Hardware



Quantum computer hardware development

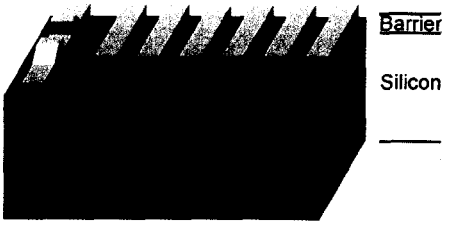
How do you Build a Quantum Computer?

- **What is required to make a quantum computer?**

Requirement	Explanation
Qubits	There are quantum states that can serve as qubits
Initialization	All qubits can be placed in a standard starting state
Static Memory	Qubits must not change during storage
Unitary Operations	Arbitrary unitary operations performed on arbitrary subsets of qubits
Conditional Operations	You can make the operation performed on one qubit depend upon the state of another qubit
Readout	The value of any qubit must be accessible via some quantum measurement operation
Isolation	Qubits must not interact with their environment in between readouts
Error Correction	Unknown (and unknowable) errors can be corrected

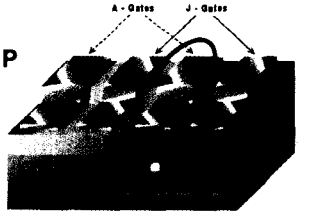
Silicon-Based Quantum Computer

- **Currently one of the most promising schemes**
  - –Invented by Bruce Kane 1998
  - –B. E. Kane, "*A silicon-based nuclear spin quantum computer*," Nature **393**, 133 (1998).
  - –Combines ideas from NMR quantum computing with semiconductor physics
  - –Builds on $1 trillion investment in Silicon technology
- **Core idea is a repetitive device structure made from Si substrate with ^{31}P dopants, insulators, metal electrodes, and RF-SETs for readout**

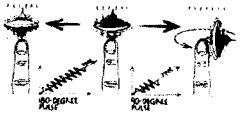


Silicon-based Quantum Computer (cont'd)

- **A qubit is the nuclear spin of a Phosphorus ^{31}P dopant atom buried inside the Silicon crystal lattice**

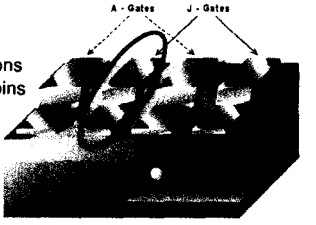


- **1-Qubit Gates**
  - At 100mk outer electron in phosphorus is loosely bound to ion
    - This electron has spin
    - This spin can influence the nuclear spin
  - By applying a voltage to one of the metal "A" gates, can change the overlap of the electron spin with its "host" nuclear spin (the qubit)
    - i.e., "A" gates control the hyperfine interaction between the electron and nucleus
  - Electron-nuclear spin interaction that determine the relative energies of the two nuclear spin states



  - This interaction selects radio frequency needed to flip the qubit on a specific P ion
  - Hence perform 1-qubit gate operations

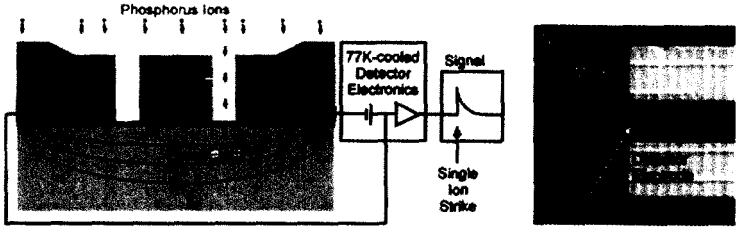
Silicon-based Quantum Computer (cont'd)



- **2-Qubit Gates**
  - "J" gates influence the overlap of electrons in neighboring ions
  - Mediates indirect coupling between two adjacent nuclear spins by controlling the exchange interaction between them
  - Allows quantum logic operations to be performed
- **Readout**
  - Map a ^{31}P nuclear spin (i.e., qubit value) to spin of an electron pair
  - Usually all electron spins aligned with external field, but apply a big "J" gate voltage, that pair of electron spins can become anti-aligned with each other
  - Whether they anti-align or stay aligned depends on whichever ^{31}P nuclear spin is most strongly coupled to its electron
  - This coupling strength controlled by "A" gate voltage
  - If spins are aligned, Pauli Exclusion Principle prevents both electrons from joining same ^{31}P ion
  - But if the electrons are anti-aligned they can
  - Circumstances distinguishable by measuring capacitance between neighboring "A" electrodes
  - Hence can "read-out" a selected nuclear spin at will

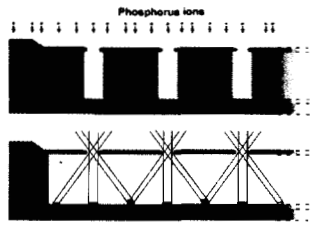
How to Fabricate such a Quantum Computer?

- **Use nanofabrication technique of ion implantation**
  - Accurate placement aided by on-chip detector technology with single ion sensitivity
  - Center for Quantum Computation, Sydney, Australia
- **Step 1: e-beam lithography makes nanoscale ion implantation masks**
- **Step 2: masks guide Phosphorus ions into Silicon substrate**
  - Arrival of each ion is detected via the electronic transient it induces in the sample
  - Ensures exactly one ion per site and facilitates device calibration

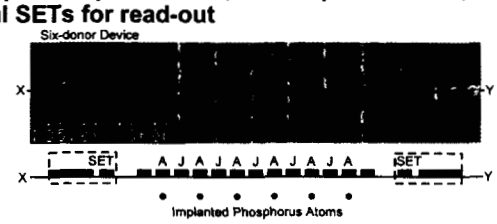


Fabrication (cont'd)

- **Step 3: use triple angle shadow evaporation to produce self-aligned gate and SET structures with "A" gates directly above implanted donors**



- **Result: complete 6-qubit device, with implanted ions, A and J gates and terminal SETs for read-out**

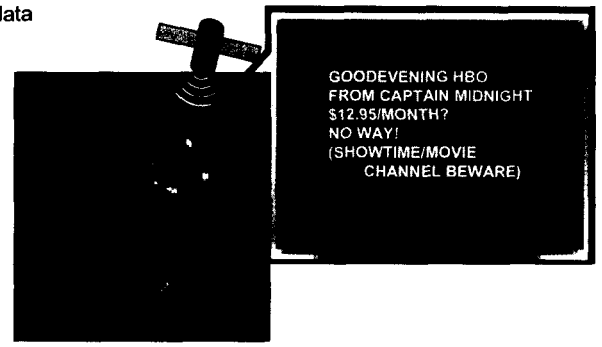


Addressing Communications Challenges

Securing Command and Control of Orbital Assets

Need for Cryptography in Space?

- **April 27, 1986 HBO satellite television broadcast briefly taken over by hacker "Captain Midnight"**
- **Highlighted vulnerability of orbital assets**
  - Need to ensure security of up-linked command paths
  - and down-linked data



- **Current solution: public key cryptography**
  - RSA, PGP or Elliptic Curve (ECC)

Is Public-Key Sufficient?

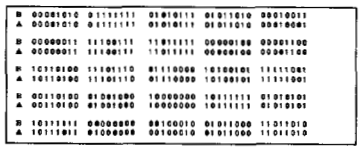
- **Security depends on *presumed* difficulty of factoring integers (RSA) and computing discrete logarithms (ECC)**
- **Could be misguided ...**
  - Factor larger integers using collaborations
    - H. J. J. te Riele, "*Factorization of a 512-bits RSA key using the Number Field Sieve*," sci.crypt posting, August 27, 1999
  - Special purpose factoring engines
    - *A. Shamir, "Factoring large numbers with the TWINKLE device" (1999)*
  - Hard to assess adversary's future computational capabilities
    - In 1977, supposed to take more than age of Universe to factor RSA-129
    - Factored in 1994 in 8 weeks by a collaboration of 1000 people
- **Recorded transmissions retro-actively vulnerable before security lifetime is reached**

- **Worse: if quantum computers become feasible all public key systems will become insecure**

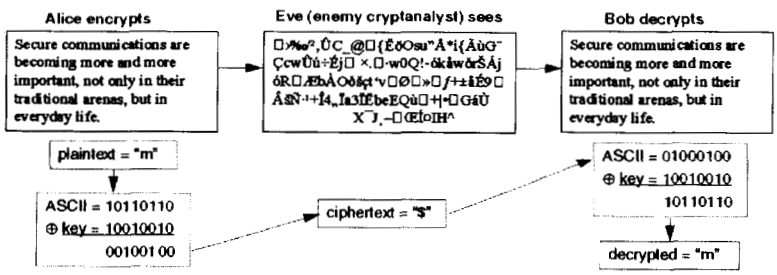
Better Solution: One Time Pad

- **Need two identical "pads" of true random numbers**

Sample of key material



- **Alice** and **Bob** generate shared key material (random numbers) using **single photon transmissions** of quantum cryptography
- **e.g.** use of key for **"one-time pad" encryption/decryption of short messages:**



Why Isn't One Time Pad Commonplace?

- **To ensure security, must never re-use keys**
  - Initial keys consumed quickly
  - So need to re-key in secure fashion
- **Re-Keying?**
  - Cannot re-key using public communications
  - Transmissions susceptible to passive eavesdropping
  - Must use trusted couriers (physical transport)

- **Couriers = security loophole**
- **Impractical for some applications**
  - e.g., satellite systems

So One Time Pad impractical because of difficulty of performing secure key distribution

Quantum Key Distribution

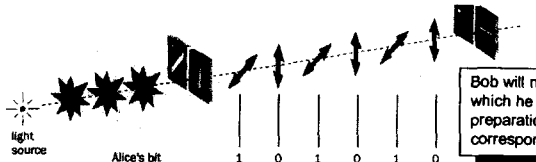
- **QKD is a scheme for establishing a common random sequence of bits between two parties**
  - By a process of single photon transmissions and classical communications
- **QKD unconditionally secure regardless of ...**
  - Computing power of adversary
  - Mathematical knowledge of adversary
  - Algorithmic advances of adversary
- **Security assured by the Laws of Nature**
  - Heisenberg Uncertainty Principle
    - Impossibility of measuring two non-commuting observables with the same apparatus without necessarily disrupting one of them
    - E.g., non-orthogonal polarization bases for photons
  - Quantum "No-cloning" theorem
    - Impossibility of copying an unknown quantum state exactly

How QKD Works (B92)

Step 1: Alice generates a secret random bit sequence

Step 2: For each bit, Alice prepares a polarized photon and sends it to Bob (in *V* = 0 / +45 = 1 encoding)

Step 3: Bob measures polarization (in –45 = 0 / *H* = 1 encoding)



Bob will never detect a photon for which he and Alice have used a preparation/measurement pair that corresponds to different bit values

Alice's bit	1	0	1	0	1	0
Alice's polarization	+45°	V	+45°	V	+45°	V
Bob's polarization	-45°	-45°	H	H	H	-45°
Bob's bit value	0	0	1	1	1	0
Bob's results	N	N	Y	N	N	Y

Quantum cryptography is a way of generating a shared key to encrypt and decrypt a message with absolute secrecy from a sequence of bits (row 1). In the B92 protocol, Alice has two filters that can linearly polarize photons vertically (V) or at +45°. For each photon she sends through free space, she chooses one of these filters at random (row 2). Bob has analysers that can measure photons that are polarized in the horizontal (H) or -45°. Every time he expects a photon to arrive, he selects one of the polarizers at random (row 3) that correspond to bit values (row 4). He records whether or not he detects a signal and communicates this information to Alice over a public channel (row 5). Alice and Bob only retain the bits for which Bob detected a photon and they use these as a secret key. Bob will never detect the photon if he selects an analyser that is incompatible with Alice's polarizer (columns 1 and 4). In the case where he does chose a compatible analyser, he has a 50% chance of detecting the photon (columns 2, 3, 5 and 6).

Obtaining Raw Key

Step 3 (cont'd): Bob never detects a photon unless his measurement is compatible with Alice's encoding

- Bob tests for 0, Alice sent a 1 – Bob sees nothing
- Bob tests for 0, Alice sent a 0 – 50% chance, etc.
- Overall Bob determines 25% bits in Alice's sequence

Step 4: To generate the key, Bob tells Alice (over a public channel) where in sequence he detected bits ... ... but not what those bits are

- From bit locations alone Alice knows bit values
- From polarizer orientations he used, Bob knows bit values
- Hence Alice and Bob know a shared "raw bit key"

Was Anyone Listening?

Step 5: Test for presence of eavesdropper ...

- If no-one was listening then their shared key is secure
- So examine statistics: Alice and Bob believe they know a common subset of the bits
- Sacrifice a proportion of these bits and compare their values
- In a perfect world they should all agree
- An anomalously high error rate reveals presence of eavesdropper
- In practice some legitimate error inevitable, hence ...

Step 6: Distill out the final key from remaining raw bits

- Perform error reconciliation and privacy amplification to reduce the information available to a potential eavesdropper to 1 bit or less

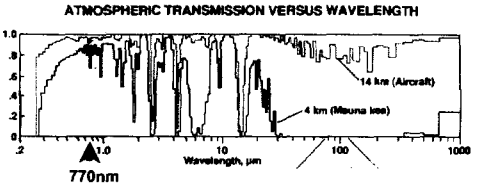
QKD Technology is Today

- **Fiber-based QKD**
  - IDQuantique (Switzerland) 70km over standard telecom fiber
  - IBM (Almaden)
  - British Telecom (U.K.) 30km fiber
  - LANL (U.S.A.) 48km over fiber
  - LANL (U.S.A.) 6-state, B92, BB84, Ekert protocols
  - Helsinki (Finland), Herriot-Watt (U.K.), Innsbruck (Austria), Oxford (U.K.), Munich (Germany), BBN (Boston), Mitre Corp.
- **Free-space QKD**
  - LANL (U.S.A.) 1.6 km daylight (now close to 10-20km)
  - QinetiQ (U.K.) 1.9 km at night

Free-Space QKD

- **Free-space (through the air) QKD proven feasible**
  - ~2km horizontally (equals more vertically)
  - Use optical window at 770nm
  - Adaptive optics to compensate for beam wander
  - Bright timing pulses, fliters, and gated detectors to eliminate background

ATMOSPHERIC TRANSMISSION VERSUS WAVELENGTH

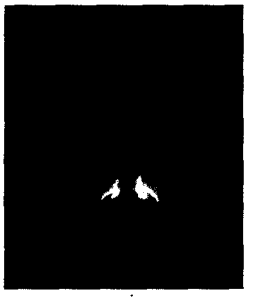


[redacted]

- **Extrapolating current data says**
  - Earth-to-satellite QKD feasible
- **Permit unconditionally secure satellite communications**

Nanotechnology for Single Photon Sources

- **Current photon source is low power semi-conductor lasers**
  - Attenuated to 0.1 photon per pulse
  - Occasionally >1 photon per pulse (not ideal)
- **Faster, true single photon sources possible**
  - Single quantum dot embedded in a semiconductor wafer with a hollow semiconductor cone mounted above dot (Yamamoto, Stanford)
  - Photon direction controllable
  - 78% efficiency
  - Single-photon "turnstiles"



Addressing Sensing Challenges

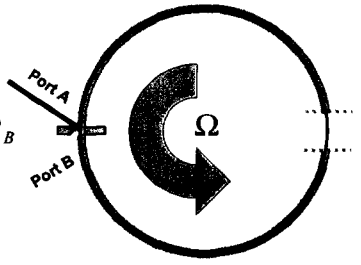
Gyroscopy

Gravitational Imaging

One-Port Fock Input

- **Exactly N photons per second in Port A and just vacuum in Port B**

$$|\psi\rangle = |N\rangle_A |0\rangle_B$$



$$\Delta\phi_{1\text{-Port}} = O\left(\frac{1}{\sqrt{N}}\right)$$

Phase sensitivity

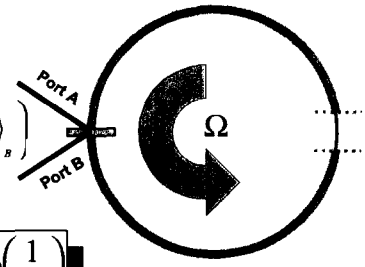
Entanglement

- **Multi-particle quantum state that cannot be factored into a definite state for each particle**
  - e.g., $|\psi\rangle = \frac{1}{\sqrt{2}}(|N\rangle_A|0\rangle_B + |0\rangle_A|N\rangle_B)$
  - Either N particles in path A and none in path B ...,
  - ... or none in path A and N in path B
  - State not definite until particle-number in a path is measured (counted)

Two-Port Entangled Fock State

- **Entangled Fock state fed into ports A and B**
- **Almost equal numbers of photons per port**

$$|\psi\rangle = \frac{1}{\sqrt{2}}\left(\left|\frac{N+1}{2}\right\rangle_A \left|\frac{N-1}{2}\right\rangle_B + \left|\frac{N-1}{2}\right\rangle_A \left|\frac{N+1}{2}\right\rangle_B\right)$$

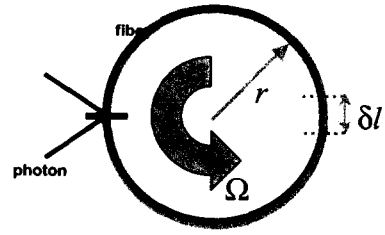


$$\Delta\phi_{2\text{-port}} = O\left(\frac{1}{N}\right)$$

Phase sensitivity

Quantum / Classical Sensitivity

- **Minimum detectable rotation rate, $\Delta\Omega$**
  - If N = total number of particles passing through device per unit time
  - ~ 10^{16} photons per sec
- **Classically,** $\Delta\Omega_{\text{one-port}} \propto \frac{1}{\sqrt{N}}$
- **Quantumly,** $\Delta\Omega_{\text{two-port}} \propto \frac{1}{N}$



- **Hence 2-port quantum optical gyro 10^8 times more sensitive to rotation than equivalent 1-port optical gyro!**

Quantum Gyroscopy Applications

- **Precise rotation sensing needed for**
  - Altitude/attitude control
  - Recovery in turbulent flight
  - Drone formation-flying
  - Inertial navigation
  - Instrument pointing & stabilization
  - Unjammable GPS
  - Autonomous vehicles
  - Covert navigation

- **Quantum gyroscope is feasible**
  - Expected to be ~ 10^6 to 10^{10} times more sensitive to rotation than existing gyros!
  - "Correlated Input-port, Matter-wave Interferometer: Quantum Noise Limits to the Atom-laser Gyroscope", J. P. Dowling, Phys. Rev. A, Vol. 57, No. 6, June (1998)

Quantum Lithography

Conventional view: feature spacing limited by wavelength of light used (Rayleigh criterion):

Spacing = $\lambda/(2 \sin(\theta))$

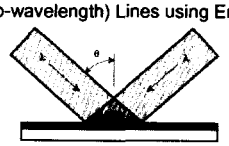
But by interfering quantum entangled photons $|0\rangle|N\rangle+|N\rangle|0\rangle$ we obtain:

Spacing = $\lambda/(2 N \sin(\theta))$

Beat Rayleigh criterion by factor of N

Interferometric Lithography

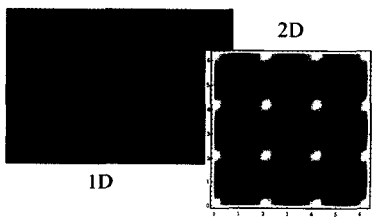
Finer (Sub-wavelength) Lines using Entangled Light



period = $\frac{\lambda}{2 \sin(\theta)}$ 136 nm at 257 nm/80°

*"Quantum Interferometric Optical Lithography: Exploiting Entanglement to Beat the Diffraction Limit", A.. Boto, P. Kok, D. Abrams, S. Braunstein, C. P. Williams, and J. Dowling, Physical Review Letters, Vol 85, 13, (2000) pp.2733-2736

Linear improvement of N gives density improvement of N^2



Currently know how to do N = 2, 3, 4 in principle, N can be arbitrarily large

Ideal for ultra-fine diffraction gratings (uses in extreme spectroscopic astronomy)

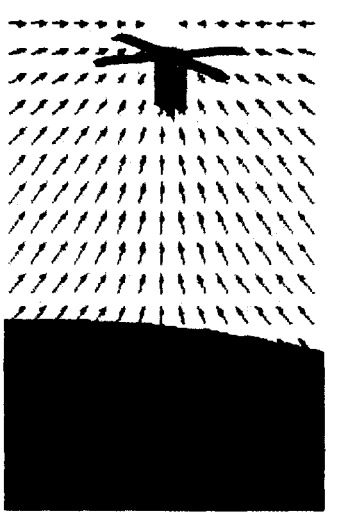
More complex 2D patterns achieved by using multiple exposures using different photon input states

Input states are "Fock states" – highly non-classical light

Quantum Gravity Gradiometry

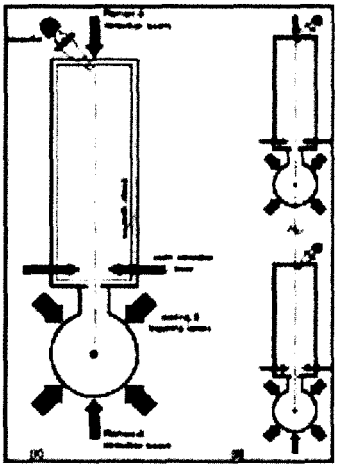
"Prospecting" from Above

- **Similar techniques can be applied to gravitational gradient sensing**
- **Applications**
  - Oil prospecting
  - Imaging interiors of planetary bodies
  - Studying lava flows
  - Imaging subterranean facilities (bunkers)



Quantum Gravity Gradiometers

- **Measure the relative acceleration of two laser-cooled ensembles of atoms**
- **Since accelerations measured for both ensembles simultaneously, vibration and spurious accelerations cancel out**
- **Better than 0.001 E/Hz$^{1/2}$**

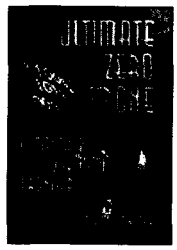


Conclusions

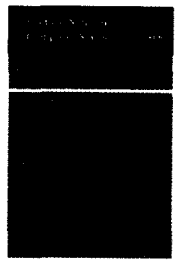
- **Dangerous to make predictions**
  - *–Where a calculator on the Eniac is equipped with 18,000 vacuum tubes and weighs 30 tons, computers in the future may have only 1000 tubes and weigh only 1½ tons (Popular Mechanics, March 1949)*
- **Advantages of nanoscale quantum computers**
  - –Scalability
  - –Integration with conventional microelectronics
  - –Size, mass, power
- **Quantum computers can solve IT challenges more efficiently than conventional computers**
  - –Computing
  - –Communications
  - –Sensing
- **Fabricate requires nanofabrication techniques**
  - –Ion implantation
  - –STMs
  - –Self-assembly
- **With just 50 qubits can simulate physical systems beyond the reach of any conceivable supercomputer**

Further Reading

Book + CD-ROM containing Simulations of Quantum Algorithms and Toolkit for Manipulating Quantum Information



Overview of Quantum Computing and Related Technologies



Proceedings of First NASA Conference on Quantum Computing and Quantum Communications

- Contact Colin.P.Williams@jpl.nasa.gov