Quantum Computation and Communication

Tom Lake
tswsl1989@sucs.org

16/02/2012

quan·tum me·chan·ics: The branch of mechanics that deals with the mathematical description of the motion and interaction of subatomic particles
- OED

quan·tum me·chan·ics: The branch of mechanics that deals with the mathematical description of the motion and interaction of subatomic particles
- OED

I think I can safely say that nobody understands quantum mechanics
- Richard Feynman, in The Character of Physical Law (1965)

Quantum Mechanics

- Under the Copenhagen Interpretation, quantum mechanics allows us to compute the probability of obtaining a given result from a given measurement

Quantum Mechanics

- Under the Copenhagen Interpretation, quantum mechanics allows us to compute the probability of obtaining a given result from a given measurement
- A quantum system can exist in a superposition of states, each with a different probability

Quantum Mechanics

- Under the Copenhagen Interpretation, quantum mechanics allows us to compute the probability of obtaining a given result from a given measurement
- A quantum system can exist in a superposition of states, each with a different probability
- We cannot say for certain which of these states the system is in, until we measure it

Quantum Mechanics

- Under the Copenhagen Interpretation, quantum mechanics allows us to compute the probability of obtaining a given result from a given measurement
- A quantum system can exist in a superposition of states, each with a different probability
- We cannot say for certain which of these states the system is in, until we measure it
- These properties yield some interesting possibilities for computation and communication

Some conventions:

- $|a\rangle$ - Quantum state a
- $\langle a| = (|a\rangle)^*$ - Complex conjugate of state a
- $|\psi\rangle = \alpha |a\rangle + \beta |b\rangle + \gamma |c\rangle$
  State ψ, which can measured to be a with probability $|\alpha|^2$, b with probability $|\beta|^2$ or c with probability $|\gamma|^2$
  Once measured, we get $|\psi\rangle = |a\rangle$ **or** $|\psi\rangle = |b\rangle$ **or** $|\psi\rangle = |c\rangle$
- $\langle a \mid \psi\rangle$ - The probability of measuring ψ to be in state a
  $\langle a \mid a\rangle = 1$, $\langle b \mid a\rangle = 0$, $\langle a \mid \psi\rangle = |\alpha|^2$
- $|a\rangle |b\rangle = |ab\rangle$

I intend to avoid getting too mathematical, but using this notation is much more convenient than lengthy descriptions.

For the computational side of things, we need a couple more definitions:

- $|0\rangle$ - Can be represented as a vector $\binom{1}{0}$
- $|1\rangle$ - Can be represented as $\binom{0}{1}$
- $|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$ unless otherwise stated
  Can be represented as $\binom{\alpha}{\beta}$

$|\psi\rangle$ is a qubit, or quantum bit.
In order to maintain a valid quantum state, we have to impose $|\alpha|^2 + |\beta|^2 = 1$.
This ensures that the maths works out correctly and provides us with sensible probabilities
(This applies to quantum mechanics generally, not just quantum computing)

Quantum Computing

- The idea of superimposed states is key to understanding the advantages of quantum computing

Quantum Computing

- The idea of superimposed states is key to understanding the advantages of quantum computing
- We can act on qubits with a set of quantum gates and measure the result

Quantum Computing

- The idea of superimposed states is key to understanding the advantages of quantum computing
- We can act on qubits with a set of quantum gates and measure the result
- Carefully thought out, this can provide a much faster way to perform calculations than a classical method

Quantum Computing

- The idea of superimposed states is key to understanding the advantages of quantum computing
- We can act on qubits with a set of quantum gates and measure the result
- Carefully thought out, this can provide a much faster way to perform calculations than a classical method
- We can work on multiple values at once by abusing the superposition

Quantum Computing

- The idea of superimposed states is key to understanding the advantages of quantum computing
- We can act on qubits with a set of quantum gates and measure the result
- Carefully thought out, this can provide a much faster way to perform calculations than a classical method
- We can work on multiple values at once by abusing the superposition
- Qubits also have phase information associated with them

Quantum Circuits

- We can work with circuits made of quantum gates to plan a quantum computer
- Quantum gates are unitary transformations that act on qubits

Quantum Circuits

- We can work with circuits made of quantum gates to plan a quantum computer
- Quantum gates are unitary transformations that act on qubits
- A property called "Universality" states that any complex gate can be approximated using a small number of simple, single cubit gates and controlled NOT (CNOT) gates

Quantum Circuits

- We can work with circuits made of quantum gates to plan a quantum computer
- Quantum gates are unitary transformations that act on qubits
- A property called "Universality" states that any complex gate can be approximated using a small number of simple, single cubit gates and controlled NOT (CNOT) gates
- A CNOT gate has 2 inputs - *data* and *control*
- If the *control* qubit is $|1\rangle$ then the output is NOT *data*

What can we use for qubits?

There are a few methods that have been tested:

- Trapped ions

What can we use for qubits?

There are a few methods that have been tested:

- Trapped ions
    - Hard to prepare
    - Tricky to interact multiple qubits
    - Reasonably simple to read results
- Nuclear Magnetic Resonance

What can we use for qubits?

There are a few methods that have been tested:

- Trapped ions
  - Hard to prepare
  - Tricky to interact multiple qubits
  - Reasonably simple to read results
- Nuclear Magnetic Resonance
  - Very low signal to noise ratio
  - Very difficult to prepare an initial stated
  - Easier for qubits to interact (compared to ion traps)
  - Has been used to factorise numbers

- Non linear optics

- Non linear optics
  - Easy to encode and prepare states (polarisation)
  - Most single qubit gates created from phase shifters and beam splitters
  - CNOT gates are created using Kerr materials

- Non linear optics
  - Easy to encode and prepare states (polarisation)
  - Most single qubit gates created from phase shifters and beam splitters
  - CNOT gates are created using Kerr materials
  - The downside is that these materials have only a weak effect or are very absorbant
- Quantum Dots
- Diamonds

Fast searches

- Take an unsorted database of N items $\{|x\rangle\}$

Fast searches

- Take an unsorted database of N items $\{|x\rangle\}$
- Each entry consists of multiple qubits - e.g $|00110101010010\rangle$
- The entry contains multiple fields, we're looking for a match in one of those fields - e.g $|0011xxxxxxxxxx\rangle$

- Start with a state is a superposition of the whole database: $|s\rangle = \frac{1}{\sqrt{N}} \sum_{i=0}^{N} |x_i\rangle$
- Check if it matches what we're searching for - the gate for this returns -1 if the entry matches, and 1 otherwise $O = 1 - 2 |\omega\rangle \langle\omega|$
- Apply a quantum gate $U = 2 |s\rangle \langle s| - 1$
- Applying these two gates successively moves us closer towards the desired answer

- Start with a state is a superposition of the whole database: $|s\rangle = \frac{1}{\sqrt{N}} \sum_{i=0}^{N} |x_i\rangle$
- Check if it matches what we're searching for - the gate for this returns -1 if the entry matches, and 1 otherwise $O = 1 - 2|\omega\rangle\langle\omega|$
- Apply a quantum gate $U = 2|s\rangle\langle s| - 1$
- Applying these two gates successively moves us closer towards the desired answer
- $UO|s\rangle = (2|s\rangle\langle s| - 1)(1 - 2|\omega\rangle\langle\omega|)|s\rangle$
- $UO|s\rangle = (2|s\rangle\langle s \mid s\rangle + 2|\omega\rangle\langle\omega \mid s\rangle - |s\rangle - 4|s\rangle\langle s \mid \omega\rangle\langle\omega \mid s\rangle)$
- $UO|s\rangle = \frac{2}{\sqrt{N}}|\omega\rangle + \left(1 - \frac{4}{N}\right)|s\rangle$

What does that all mean?

- Every iteration moves us closer to the answer
- At any given time, if we measure the system we have a chance, $\langle \omega \mid s \rangle$ of measuring the correct answer

What does that all mean?

- Every iteration moves us closer to the answer
- At any given time, if we measure the system we have a chance, $\langle \omega \mid s \rangle$ of measuring the correct answer
- $\sqrt{N}$ iterations gives us the best chance of measuring the right result
- But as we can easily check for the right result, we can run the search again in the rare case that we get the wrong result

What does that all mean?

- Every iteration moves us closer to the answer
- At any given time, if we measure the system we have a chance, $\langle \omega \mid s \rangle$ of measuring the correct answer
- $\sqrt{N}$ iterations gives us the best chance of measuring the right result
- But as we can easily check for the right result, we can run the search again in the rare case that we get the wrong result
- The probability of measuring the wrong answer decreases as the size of the database increases

So we now have an $O\left(\sqrt{N}\right)$ algorithm for an unsorted database

All your RSA are belong to us

- RSA uses two large prime numbers (p, q) to generate part of the private key (N)
- The fact that N has only these two factors is important!

All your RSA are belong to us

- RSA uses two large prime numbers (p, q) to generate part of the private key (N)
- The fact that N has only these two factors is important!
- A message encrypted with RSA (by using the public key) can be decrypted with the corresponding private key
- Without the private key, the unencrypted message can be recovered by either:

All your RSA are belong to us

- RSA uses two large prime numbers (p, q) to generate part of the private key (N)
- The fact that N has only these two factors is important!
- A message encrypted with RSA (by using the public key) can be decrypted with the corresponding private key
- Without the private key, the unencrypted message can be recovered by either:
  - Brute force - encrypt every possible message using the public key until you find a match

All your RSA are belong to us

- RSA uses two large prime numbers (p, q) to generate part of the private key (N)
- The fact that N has only these two factors is important!
- A message encrypted with RSA (by using the public key) can be decrypted with the corresponding private key
- Without the private key, the unencrypted message can be recovered by either:
  - Brute force - encrypt every possible message using the public key until you find a match
  - Factorise N into it's two factors, enabling you to calculate the private key

All your RSA are belong to us

- RSA uses two large prime numbers (p, q) to generate part of the private key (N)
- The fact that N has only these two factors is important!
- A message encrypted with RSA (by using the public key) can be decrypted with the corresponding private key
- Without the private key, the unencrypted message can be recovered by either:
  - Brute force - encrypt every possible message using the public key until you find a match
  - Factorise N into it's two factors, enabling you to calculate the private key
- Factorising large numbers takes classical computers a long time!

Shor's Algorithm

- Shor's algorithm reduces the factorisation of a large number N to the problem of finding the period of a function
- $\mathrm{f}(x) = a^x \mod N$
- $a < N$ and $\gcd(a, N) = 1$

Shor's Algorithm

- Shor's algorithm reduces the factorisation of a large number N to the problem of finding the period of a function
- $\mathrm{f}(x) = a^x \mod N$
- $a < N$ and $\gcd(a, N) = 1$
- Use a quantum Fourier Transform to find the period r of f
- The two prime factors of N are then given by $\gcd\left(a^{\frac{r}{2}} \pm 1, N\right)$
- This runs in $\mathrm{O}((\log N)^3)$, rather than the exponential time required classically

Any questions?

Slides will be available via talks section of the SUCS website, or at
`http://sucs.org/~tswsl1989/talks/`