

# Quantum Computing 2: Applications

## Lecture 22 – 4/14/04

Charles Vitu

### Readings for this lecture:

- “Unconditional Security of Quantum Key Distribution over Arbitrarily Long Distances,” by Hoi-Kwong Lo and H. F. Chau. Appeared in Science, March 26 1999. (1)
- “Quantum Algorithm for Clock Synchronization,” by Isaac L. Chuang. Appeared in Physics Review Letters, August 2000. (2)
- “Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer,” by Peter W. Shor. Expanded version of a paper from the 35th Annual Symposium on Foundations of Computer Science, 1994. (3)

### Lecture Overview:

This lecture summarizes three applications, which utilize the special properties of quantum physics to complete tasks in a way that is fundamentally impossible with classical methods.

### Quantum Key Distribution:

The art of cryptography has been around for a long time. Truly secure communications between two parties is only really possible if both parties can share a random string of bits. One-time pad cryptography has been proven to be impossible to break, yet is amazingly simple. You take your message and perform a bitwise XOR with a random number string, called a one-time pad. The resulting message can be safely transmitted publicly.

	10010110	Secret Message
XOR	10111011	Random Number String
	-----	
	00101101	Transmitted Message

This form of encryption is truly unbreakable, if you realize that there are as many possibilities for a one-time pad as there are messages to be sent. However, there are some problems with it. The first problem is getting the random bits to both the sender and the receiver. If the transmission of the one-time pad is message is observed on route to either party, then security of the communications will be compromised. The second problem is assuring true randomness of the one-time pad bits.

Quantum Key Distribution (QKD) uses the special property of entanglement to resolve both of these issues in a way that was never before thought possible, using classical physics. To transmit one bit of the one-time pad, an Electron Paramagnetic Resonance pair, an EPR pair, is generated. An EPR is a pair of electrons in a superposition of spin states. The pair is separated, with one given to the sender and one to the receiver.

Randomness – Sender and receiver both measure their particles for spin orientation. Since the particles are in a superposition state, their measurement locks in the state of the EPR in an essentially random fashion.

Key Distribution - The quantum no-cloning theorem states that it is impossible to make an exact copy of an unknown quantum state (1). In effect, the act of observing the quantum state alters it. This allows both the sender and receiver to determine if anyone is listening to the quantum channel. In order to ensure that no one is listening in, both the sender and receiver compare an arbitrary number of the sent bits. If enough bits are compared and match, then both parties can be assured with certainty, that there is an extremely low probability that no one was listening. The remaining bits are then used to encrypt the message.

Issues with QDK - The problem with entangled electrons is that they often have a difficult time staying entangled. Collisions with the transport medium, and time itself can lead to what is called decoherence. Decoherence causes an EPR pair that would otherwise be entangled (i.e. it hasn't been observed), to become disentangled. This effect introduces unwanted noise into the system. As was previously mentioned, having the sender and receiver compare an arbitrary

number of bits in the sent key ensures the integrity of the quantum communication channel. However, if the entanglement between the EPR pairs is broken due to decoherence, the sender and receiver will have mismatches when comparing bits of the key, even if no one is eavesdropping on the quantum channel. The paper presents two solutions to this problem. One way is to calculate an acceptable number of mismatches in the comparison process. However, if the communication distance is extremely long, then the decoherence problem may be too large to overcome in this fashion. In this case, the author suggests the use of quantum repeaters (1). These are essential quantum computers that divide the quantum channel into shorter segments.

QDK is one of the few quantum computing applications which has been shown to work experimentally. It was, in fact, recently shown to work over a non-trivial distance of several kilometers (1). This technology could prove to be important especially if another quantum computing application, which is discussed in the next section, can be further developed.

### **Quantum Clock Synchronization:**

Clock synchronization is extremely important for several technologies. Navigation through GPS, global communications, and electrical power generation all rely on precisely synchronized clocks (2).

The goal is to have two synchronous clocks in arbitrary locations. We can try to do this classically in one of two ways.

One approach would be to synchronize the two clocks in one location and then move one away from the other. The problem arises when the two clocks are then moved away from each other. In order for this to happen one must be accelerated. Einstein's theory of relativity dictates that the accelerated clock will experience what is called time dilation. Even the relatively small acceleration experienced on an airplane is enough to throw the two clocks out of sync. This time difference, called delta, is very small, perhaps only a few nanoseconds.

A different approach is for one clock to broadcast its current time. The second clock can observe this time, add the transit time for the signal and thus compute the time difference. However, this approach doesn't work because the delivery time of that message is not guaranteed to be consistent.

In his paper, Isaac L. Chuang proposes a quantum algorithm to precisely synchronize two clocks, independent of delivery time (2).

Say we have two people, Alice and Bob, both of whom have clocks they wish to synchronize.

1. Alice sends Bob three things:

1. The current time at her location  $t_{a1}$
2.  $\omega$ , the rate at which the particle's phase is changing.
3.  $\psi$ , the particle itself.

Note that both  $t_{a1}$  and  $\omega$  can be sent classically.

2. Bob receives the particle at  $t_{b2}$  which has now under-evolved into a state  $e^{i\omega(\text{transit time})Z}$  times its original state.

3. Bob then transforms the particle by  $Xe^{-i\omega(t_{b2}-t_{a1})Z}$  giving him

$$Xe^{-i\omega(\text{delta})Z} * \psi$$

4. Bob then sends this particle back to Alice who does a similar transformation to get  $Xe^{-2i\omega(\text{delta})Z} * \psi$

5. Alice then observes this particle, and it will be in state 0 with a probability of  $\cos^2(2\omega(\text{delta}))$

This process can be repeated allowing Alice to obtain delta to an arbitrary accuracy.

In theory this algorithm works because the quantum particle itself acts as a time piece. The phase change of its superposition state,  $\omega$ , is not effected by acceleration.

### **Shor's algorithm:**

Shor's algorithm is often described as the Holy Grail in the quantum computing field. Shor's algorithm, which relies on quantum physics, is a method for factoring large numbers in polynomial time. The reason so much attention has been given to Shor's algorithm is that it directly attacks a principle behind the most widely used method for encrypting data.

There are several functions for which computing  $f(x)$  is far easier than computing  $f^{-1}(x)$ . For instance, most 4<sup>th</sup> graders could multiply  $4231753 \cdot 4231753$  without a calculator in an hour or so. However, most adults given a week can't compute the square root of 17907733453009. The RSA encryption algorithm is based on the fact that it's easier to multiply two prime numbers than it is to figure out what two prime factors a particular composite (non-prime) number is the product of. Thus, cracking an RSA encrypted cipher involves factoring a large number, which by classical methods requires exponential time. The best known method involves something akin to trying all the possibilities. As a result, the 128-bit RSA encryption scheme commonly used today is widely considered to be virtually unbreakable, given the computing resources currently available.

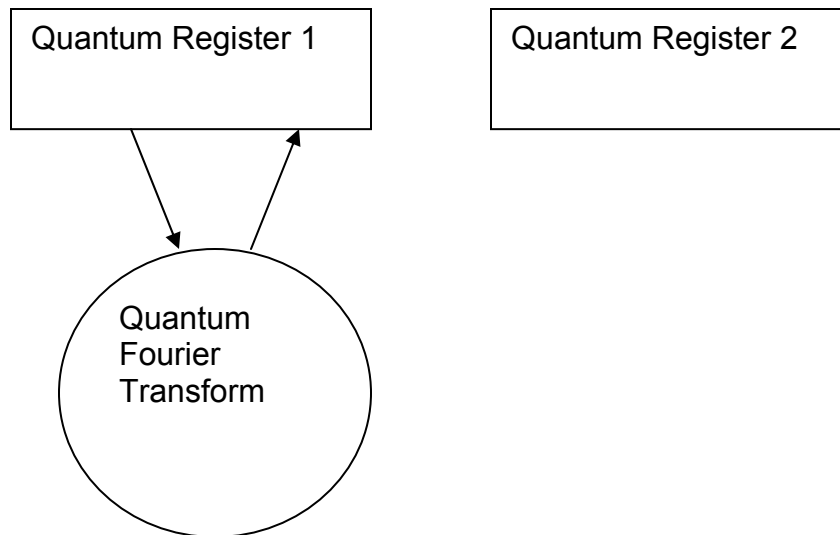
Shor's algorithm uses quantum computing and offers a method for factoring numbers that requires polynomial time to run.

### Mathematical Background:

A number  $x$  has order  $r$  if:  $x^r = 1 \pmod{n}$ , and is the smallest integer for which this is true.

Given a number  $n$ , a random number  $x$ , and  $r$ , there exists an algorithm, which runs in polynomial time that has a 50% chance of finding the factors of  $n$ . Different values of  $x$  can be used if unsuccessful. Thus all that is needed to determine factors of  $n$ , is to determine  $r$ . This is where quantum physics comes into play.

Hardware required:



There are three quantum components needed to compute the value of  $r$ . These are two quantum registers analogous to regular binary registers except that a quantum register holds qubits as opposed to bits, and a method for computing the Quantum Fourier Transform.

The first step for determining  $r$  is to compute a value called  $L$ .  $L$  is the power of two between  $n^2$  and  $2n^2$ . If  $n = 55$ , for example, then  $L = 12$  because  $55^2 < 2^{12}$  and  $2 \cdot 55^2 > 2^{12}$ . The first of the two quantum registers is loaded with the superposition of all possible states representing number  $a \pmod L$ . The second quantum register holds  $x^b \pmod n$ . After the registers have been loaded, a quantum Fourier transform is performed on the first register. The

machine is then observed, thus collapsing the superposition of the qubits in the quantum registers. The system will now be in a state where the probability peaks are such that  $r$  can be determined with relatively high probability (75 %).

Unlike many theories and conjectures in the field of quantum research, this algorithm has been shown to work. Researchers at IBM have been successful in factoring the number 15 into its primes, 3 and 5. This may not sound impressive, but it does demonstrate that this algorithm can be made to work. These results, as with many aspects of quantum computing, are under a cloud of controversy. There are some physicists who claim that entanglement, which is supposedly essential for Shor's algorithm to work, can not possibly exist for the length of time required to compute the value of  $r$ . Yet  $3 \cdot 5 = 15$  nonetheless.

This application is still in the very early stages of development, and by all means is many, many years from threatening current encryption RSA. Noise and decoherence, as with any quantum system, seriously diminish the effectiveness of this algorithm. As a result, factoring arbitrarily large numbers using this method becomes more and more difficult as more qubits are needed for error correction. However, further development of this technology could eventually make RSA encryption obsolete.

## **Conclusion:**

The field of quantum physics is still in its infancy. Quantum physicists are just now beginning to understand quantum effects and their potential to change how we think about computers. Nevertheless, even in this early stage of understanding, there does appear to be several uses for the unique behavior of quantum particles. On the surface, quantum technology may appear to be cumbersome, and unpredictable. However, even the almighty transistor had a humble beginning. Who knows what the future holds for quantum computing.