Survey of Quantum Computing

Abstract
Quantum Computing is a newer field study gathering from multiple disciplines, largely Computer Science and Physics. It is hard to pin down when Computer Science began as a field and just as hard to determine for Quantum Computing. While Quantum Computers remain theoretical many people do work in the field and there are already a number of Quantum Algorithms that have been developed. This high-level survey of Quantum Computing, from the physical aspects to its algorithmic applications, will hopefully excite interest in the field.

Overview
Computer Science isn't referred to as an old domain very often yet if you were to ask someone when the field first began they'd have a hard time answering. Some may consider it to have begun as far back as 1703 when Gottfried Leibnitz developed a formal mathematical logic system using zeros and ones to represent on and off states. This later lead to George Boole's development of Boolean algebra in 1854 which allowed computational processes to be mathematically modeled, another arguable birth point for Computer Science. There are also valid reasons to consider 1937's introduction of Turing Machines as a starting point for the field or 1946's Von Neumann architecture which is still used today. We'll be starting at 1837 with Charles Babbage and Ada Lovelace's Analytical Machine and what is often considered to be the first algorithm written to be performed by a machine.

First described in 1837, the Analytical Engine was a design for a mechanical computer which allowed for conditional branches, loops, and integrated memory. It was the first ever proposed design for a Turing-Complete machine, before Alan Turing was ever even born. Although theorized in the 19th century it wasn't until 1940 that general purpose computers were actually built. Through the use of three types of punch-cards and mechanical looms a machine is able to write and read information, as well as perform loops and all other Turing-Complete machine functionality. This process used to instruct the Analytical Engine could be analogized as its Assembly or Machine code.

In 1805 a Fast Fournier Transformation algorithm was invented by Carl Fredrich Gauss. This is the oldest algorithm I can find capable of being run on a Turing-Complete machine. In 1842 an algorithm to compute Bernoulli numbers, designed specially for the Analytical Engine, was created by Luigi Menabrea. In 1843 it was translated into English by Ada Lovelace. The modern programming language Ada is named in her honour.

Most modern computers employ the Von Neumann architecture which at a high level is simply the following. Input is fed to a processing device capable of performing controls and arithmetic in relation to memory. Output is computed as a result of an input. Memory is stored as a series of bits (zeros or ones) which can occupy a single state at any given point. The processing device is composed primarily of ALU(Arithmetic Logic Units) capable of executing a single instruction per physical thread of execution at any given point. HDDs are a form of mechanical magnetic storage. Flash Memory is a form a logic-gate based memory. ALUs are composed primarily of logic gates which are themselves made from transistors and diodes.
Quantum Computing is one of the newest sub-disciplines of Computer Science and its birth can be just as hard to pin down. 1982 with Richard Feynman's proposal of a 'quantum computer' is a definite candidate but you could argue that it couldn't be considered a field until Peter Shor developed an algorithm at the Bell Laboratories allowing a quantum computer to factor numbers far larger than on a classical computer.

In a general sense quantum superposition refers to a principle stating in any quantum system the existence of certain relations amongst states. Quantum entanglement refers to a physical phenomenon which takes place when multiple particles are generated or interacting in a manner where their quantum state cannot be described independently.

Qubits are the quantum counterpart to a bit. While a classical bit can only be in one of two possible states a qubit can be in a superposition of states. It can be thought of as achieving both states at once. The possible states of a single qubit can be visualized using a Bloch sphere.

Quantum gates are the quantum counterparts to logic gates. While most logic gates are not reversible, all quantum gates are reversible. All forms of classical computing are possible on reversible gates so a quantum computer can still perform classical computing. Similar to how most logic gates operate on one or two bits, quantum gates operate on one or two qubits. Quantum gates are represented as unitary matrices.

A quantum algorithm is an algorithm which runs on the quantum circuit model of computation. A classical algorithm is a comprised of a finite sequence of procedural orders for solving a problem. Quantum algorithms are also a sequence of procedural orders. A quantum computer or quantum algorithm can not be used to solve a problem which is not possible with a classical computer or classical algorithm. This means problems which are undecidable on a classical computer remain undecidable on a quantum computer.

Quantum complexity classes exist alongside the classical complexity classes such as P and NP. BQP is a quantum complexity class analogous to P representing the set of problems solvable by a quantum computer in polynomial time with bounded error. QMA is a quantum complexity class analogous to NP representing the set of decision problems whose solution or proof is verifiable on a quantum computer in polynomial. The verifier accepts correct proofs for YES whose probability is greater than 2/3. If the answer is NO then there is no proof the verifier accepts with probability less than 1/3. These constants can be changed. Where quantum complexity classes lie in relation to their classical counterparts remains one of the main goals of quantum complexity theory.

A quantum algorithm in the quantum complexity class BQP used for integer factorization. RSA relies on the hardness of factorization so Shor's Algorithm renders it useless with a quantum computer. DSA relies on the hardness of discrete logarithm which a variation of Shor's Algorithm renders useless with a quantum computer. Symmetric encryption and hash algorithms are weakened as well but not to the same level as RSA and DSA.

Shor's Algorithm is used to find the prime factors P and Q of an integer N. Imagine two sine waves of lengths P and Q. Assuming P and Q are co-prime then we can try to find the point where the harmony of P and Q overlapped repeats itself. If we look at point N then the phase of P and Q must be 0 if they are factors of N. Since phase estimation is for quantum computers, Shor's Algorithm makes factorization easy.
References


