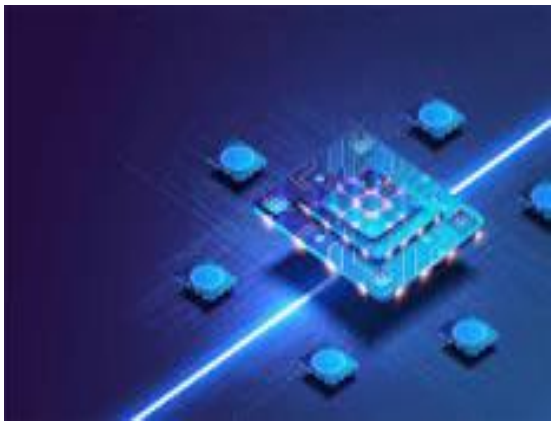
SUMMER OF SCIENCE

QUANTUM COMPUTING AND INFORMATION

SNEHAA REDDY(210070067)

MENTOR: ANAGHA BHANGARE



1: INTRODUCTION TO QUANTUM COMPUTING

- History of quantum computation
  Quantum computing evolved from the ideas of Feynman. He realised that if we simply consider the particles following their natural quantum mechanical behaviour as a computer then this “quantum computer ” appears to be performing this computation exponentially faster than that done by “classical” ones.

- Boolean circuits
  Here we are interested in coming up with efficient Boolean circuits for every operation. Circuits with smaller sizes have greater efficiency. The number of gates compares to the size of a circuit. The running time of an algorithm is dependent on the number of gates.

- Reversible computation
  Classical gates such as AND and OR gates are not reversible since the inputs cannot be found given the outputs thus some form of “entropy” is being lost here.
  It is possible to construct universal, reversible gates which do not dissipate any energy if they have an equal number of inputs and outputs.
  Ex: NOT, CNOT, CCNOT

- Measurement of quantum states
  The probabilistic “state” of the registers at some intermediate time reflects the uncertainty about the quantum states. Analytically, once we observe one or more of the r-bits, the probabilistic state “collapses” to reflect the state of the system.

- Quantum computation
  Here we introduce the Hadamard gate. In quantum computation we describe the state of an n- qubit system as a linear combination of its 2^n base states. The coefficients of each state can be any real or

complex number. The square of the modulus of each coefficient gives the probability of that state. Also the sum of all these probabilities should be 1.

- On measurement
  In randomised computation the probabilistic state of the registers only represents the uncertainty of the observers. They actually have a determined state. However this is not the case with quantum computation.
  According to the laws of physics, in the middle of a quantum circuit's computation, the superposition state that the n qubits are in literally the true state they're in Nature. They are not secretly in one of the basic states; Nature is literally keeping track of the 2n amplitudes.

2: QUANTUM MATH BASICS

- Complex numbers
  Complex numbers are an essential part of quantum computing. They help us to contemplate a lot of things in a very easy way.

- Quantum bits
  Just as a classical bit can have a state of either 0 or 1, the two most common states for a qubit (quantum bit) are the states ket(0) and ket(1). However, a qubit can be in a state of linear combination with the sum of modulus squares of the coefficients equal to 1.

- Multiple qubits and qudit system.
  A state in the d-dimensional qudit system is a superposition of d basis states. In reality, particles can have spin quantum number of 0,1/2,1,3/2 etc. A spin half particle like a electron is a "qubit"system, whereas a spin 1 particle is a "qutrit" system.

- Qubits- the mathematics
  A quantum state in the qubit system can be represented as a unit column vector in the complex plane. A quantum state is a column vector, also known as a ket, whereas a state in the row vector dual to, also known as a bra. To get a bra vector from a ket vector we need to take the transpose conjugate of the later.
  The inner product of two quantum states is given by the multiplication of the ket form of the first vector with the bra form of the second vector. I also learned about certain properties of inner and outer products of two vectors.

- Multiple Qubit System
  Tensor products are exactly like vector spaces in linear algebra. The states that cannot be expressed in terms of a tensor product of two other states are called the entangled states.

- Quantum computation
  The real number analogue of unitary matrices is the orthogonal matrix. Unitary matrices are those with the property U*U(dagger) = I, where U(dagger) is the complex conjugate transpose of the matrix U. Unitary operations preserve the angle between two quantum states. Unitary operations are invertible. The inverse of unitary matrices is their complex conjugate transpose.

3: Quantum Entanglement

- ### No-cloning theorem

  To start, we describe a way in which quantum computation is not superior to classical computation. In classical computation, it is quite easy to copy information, even in a reversible manner. Given a qubit psi is it possible to copy it so that two untangled qubits have the same state psi. This would require the two qubits to be in a joint state of psi ⊗ psi. The theorem states that:
  For all n ∈ N, there exists no quantum circuit C which upon input psi ⊗ 0^(n-1) outputs psi ⊗ psi ⊗ f(psi), where f(psi) (the garbage) is a possibly entangled state of n − 2 qubits.

- ### EPR pairs

  The state $1/\sqrt{2}(00) + 1/\sqrt{2}(11)$ is an entangled state called the EPR pair. This pair is named after Einstein, Podolsky, and Rosen. Although Einstein did not believe that EPR pairs existed, they have been confirmed through experimentation.
  When we pass an EPR pair $1/\sqrt{2}(00) + 1/\sqrt{2}(11)$ through a quantum gate H⊗H, that is we apply the Hadamard gate to each quantum bit, then we get the result as the EPR pair itself.

- ### Quantum Teleportation

  Imagine that two computer scientists, Alice and Bob, each have a qubit initialized to the classical state |0i and that they decide to entangle their qubits into an EPR pair. So, their joint state is $1/\sqrt{2}(00) + 1/\sqrt{2}(11)$. The first qubit is Alices's and the second is Bob's. Now, even if their qubits are physically separated the two qubits will still be an EPR pair as long as neither performs a measurement. Now Alice can give a qubit psi to Bob without going anywhere, thus they can perform teleportation. This can be done using an EPR pair and two classical bits. This is not a violation of the "no cloning" theorem as Alice no longer has the copy of her quantum bit.
  It turns out that this theory called "quantum teleportation" not merely works in theory but it has been verified to work in practice. One drawback though to quantum teleportation is that it does not help

someone ‘believe’ in quantum mechanics. That is, in order to interpret the results of these experiments one needs to already accept quantum mechanics.

- Rotating the basis

  Often, the basis we which to measure in is a counter-clockwise rotation of the standard basis by an angle of θ. Thus, the change of basis matrix we desire for measuring is the clockwise rotation matrix by the angle of θ, which we denote as Rotθ

$$\text{Rot}\theta = \begin{matrix} cos & sin \\ -sin & cos \end{matrix}$$

  Lemma：Let θ and γ be angles and let Δ = θ − γ. Then,
  (Rotθ ⊗ Rotγ)(1/√2(00) + 1/√2(11)) = 1/√2 (cos Δ(00) + sin Δ(01) − sin Δ(10) + cos Δ(11)).

  Note that is θ = γ, then Δ = 0, and the resulting state is the EPR pair.

  Corollary: After applying Rotθ ⊗ Rotγ to an EPR pair, perform a measurement. The probability that both qubits in the collapsed state have the same value is cos^2 Δ = cos^2(θ −γ). Likewise, the probability that both qubits collapse to different states is sin^2 Δ.

- CHSH game

  The CHSH Game consists of a team of two players Alice and Bob who are assisted by two referees Ref. 1 and Ref. 2. Alice and Bob are separated sufficiently far away (say 1 light-second) so that they cannot communicate with each other during the game, but Alice is sufficiently close to Ref. 1 and Bob is sufficiently close to Ref. 2. At the start of the game, Ref. 1 and Ref. 2 pick select uniformly random bits x, y ∈ {0, 1}, respectively. Ref. 1 tells Alice x and Ref. 2 tells Bob y. Alice is then to respond with a bit a ∈ {0, 1} and Bob is to respond with a bit b ∈ {0, 1}. Alice and Bob win if and only if a ⊗ b = x ∧ y.

  1. Classical strategy

It is easy to come up with a strategy in which Alice and Bob win with a probability 0.75 : have both players always respond with 0. It turns out that no classical strategy can do better. In fact, no classical strategy can do better than this.

2. Quantum strategy
   For this quantum strategy to work, we will have Alice and Bob each share a qubit of an EPR pair. Alice and Bob will independently decide which basis to measure their qubit in the EPR pair based on the random bit they receive. By exploiting the correlations of the EPR pair, Alice and Bob will get a win probability significantly greater than 0.75.
   Theorem: There is a quantum strategy which allows Alice and Bob to win with probability $\cos^2(\pi/8) \approx 0.85 > 0.75$.

   Like with quantum teleportation, multiple experiments have been done to verify that the quantum strategy beats the classical one.

4: Introduction to quantum information

- ## Classical information theory

In classical information theory, we typically have some random variable X distributed according to P on some set M. The most basic question one can ask is how much information do we learn from seeing X?

Example: Suppose that M = {0, 1}n.
• If P is the uniform distribution, then one gets n bits of info from seeing X.
• If P has all its probability on a single string x0 belonging to {0, 1}^n, i.e. X = x0 with probability 1 and X = x with probability 0 for all x not equal to x0, then we get 0 bits of information from seeing X.

Definition: Shannon entropy
The shannon entropy of a random variable X distributed on a set M is

$$H(X) = \Sigma p(x)\log(1/p(x))$$

Where p(x) represents the probability of P(X = x).
Note that the Shannon entropy function is concave. The largest amount of information that anyone could hope to learn is H(X).
We have two correlated random variables X and Y . We want to know how much knowing Y tells us about X. In general, if we have random variables X and Y supported on the sets M and N respectively with joint distribution P(x, y) = Pr[X = x, Y = y], we have

$$H(X,Y) = \Sigma p(x,y)\log(1/p(x,y))$$

In general, we note that if X and Y are independent, then H(X, Y) = H(X) + H(Y). This seems reasonable, as seeing one of the random variables tells us nothing about the other, so seeing half of the pair (X, Y) only decreases tells us the shannon entropy of the random variable that we observe, but the other random variable still has all of its entropy.

Conversely, if X and Y perfectly correlated, then H(X, Y) = H(X) = H(Y). Indeed, seeing half of the pair (X, Y) immediately tells us what the other half of the pair is, so the amount of entropy in the pair is the same as the amount of entropy in the random variables themselves.

The mutual information I(X; Y) between two random variables X and Y is I(X; Y) = H(X) + H(Y) H(X, Y). This is supposed to represent the amount of information you learn about X from knowing what Y is. Since the definition is symmetric in X and Y , it also represents the amount of information you learn about Y from knowing X.

- Quantum information theory

The correct way to define the Shannon entropy for quantum analogue is as follows:
Definition: Given a mixed state, let ρ be the density matrix and suppose it has eigenvalues a1,a2,a3,…..,an with corresponding eigenvectors psi1,psi2,psi3,…psin. We define

$$H(\rho) = \Sigma\, ai \log(1/ai) = H(a)$$

This quantity is often referred to as the von Neumann entropy.

Definition: (Quantum Mutual Information). If ρ is the joint state of two quantum systems A and B then the quantum mutual information is

$$I(\rho A; \rho B) = H(\rho A) + H(\rho B) - H(\rho).$$

5: Quantum algorithms

- ## Introduction

  An algorithm is a step-by-step procedure to perform a calculation, or a sequence of instructions to solve a problem, where each step can be performed on a computer. Therefore, an algorithm is a quantum algorithm when it can be performed on a quantum computer. In principle, it is possible to run all classical algorithms on a quantum computer. However, the term quantum algorithm is applied to algorithms of which at least one of the steps is distinctly 'quantum', using superposition or entaglement.

- ## Quantum circuits

  A quantum circuit is a model for quantum computation, where the steps to solve the problem are quantum gates performed on one or more qubits. A quantum gate is an operation applied to a qubit that changes the quantum state of the qubit. Quantum gates can be divided into single-qubit gates and two-qubit gates, depending on the number of qubits on which they are applied at the same time. Three-qubit gates and other multi-qubit gates can also be defined. A quantum circuit is concluded with a measurement on one or more qubits.
  A difference with a classical algorithm is that a quantum algorithm is always reversible. This means that if measurements are not a part of the circuit, a reverse traversal of the quantum circuit will undo the operations brought about by a forward traversal of that circuit.

- ## Shor's algorithm

  Problems that are fundamentally unsolvable by classical algorithms (so-called undecidable problems) cannot be solved by quantum algorithms either. The added value of quantum algorithms is that they can solve some problems significantly faster than classical algorithms. The best-known examples are Shor's algorithm and Grover's algorithm. Shor's algorithm is a quantum algorithm for integer factorization. Simply put, when given an integer N, it will find its prime factors. It can solve this problem exponentially faster than the best-known classical algorithm

can. Grover’s algorithm can search an unstructured database or unordered list quadratically faster than the best classical algorithm with this purpose.

6: Grover's algorithm

Grover's algorithm solves the problem of unstructured search

Definition: In an unstructured search problem, given a set of N elements forming a set X = {x1, x2, . . . , xN } and given a boolean function f : X → {0, 1}, the goal is to find an element x* in X such that f(x*) = 1.

Unstructured search is often alternatively formulated as a database search problem in which we are given a database and we want to find an item that meets some specification. For example, given a database of N names, we might want to find where your name is located in the database. The search is called "unstructured" because we are given no guarantees as to how the database is ordered. If we were given a sorted database, for instance, then we could perform binary search to find an element in logarithmic time. Instead, we have no prior knowledge about the contents of the database. With classical circuits, we cannot do better than performing a linear number of queries to find the target element.

Theorem: The unstructured search problem can be solved in O(√ N) queries using quantum computation.

- The classical case

  We need some sort of model for how we are allowed to interact with our database. In a classical algorithm, our medium of interaction is an oracle Of that implements a function f. The function f encodes information about the element of interest, and we query the oracle to fetch results from the function. Think of the oracle as a quantum circuit or a big gate.

  This oracle, given an input of i, outputs f(i) in constant time. As we do with any good algorithm, we want to limit resources used, which in this case are running time and number of queries to the oracle.

- The modifications in the quantum case

  First, we are going to assume that the element x ∗ that satisfies f(x*) = 1 is unique. Second, we are going to assume the size of our database is is a power of two, letting N = 2n where N is the size of the database. Lastly,

we are also going assume that the data is labelled as n-bit boolean strings in {0, 1} n rather than being indexed from 1 to N. In turn, our boolean function f that we are studying will map {0, 1}^n to {0, 1}. These conditions will be more convenient for us, and we do not lose much; the algorithm can be extended to relax the uniqueness assumption, we can always add garbage entries to the database to make the size of the database a power of two, and the naming of the elements is arbitrary.

- The algorithm

Our goal is to, given the ability to query f using our oracle gate, find x $*$ such that f(x*) = 1. We start with n qubits all initialized to 0. Think of these as forming an n-bit string, or input to f. Let us make our goal concrete by saying that we would like the amplitude of x*to be at least 0.1 in magnitude at the end of the circuit so that when we measure at the end, we have a 0.1 *0.1 = 0.01 probability of measuring x*.What can we do with our quantum circuit that is not possible classically? We can perhaps exploit superposition. We shall begin by transforming our input into the uniform superposition

$$\sum 1/\sqrt{N}\ (x)$$

The uniform amplitudes of all the n-bit strings, in a sense, represents our current complete uncertainty as to what x* is. We achieve this uniform superposition of all the basis states by sticking a Hadamard gate on every wire.

Now we shall query this state by adding an oracle O(f) gate, that flips the amplitude of the x* component and leaves everything else unchanged, giving us a state of

$$-1/\sqrt{N}(x^*) \ + \ \sum 1/\sqrt{N}\ (x)$$

The second term in the above summation is for all combinations of x except for x*.We want to increase the amplitude of x *absolutely. This motivates the introduction of a new gate that performs what is called the Grover diffusion operator. What does this gate do? First, we have to define a new number

$$\mu = 1/N*\Sigma a\_x = \text{average of } a\_x \text{ for } x \in \{0, 1\}^n .$$

where a_x is the amplitude of x for a given x in {0, 1}^n. Our Grover diffusion gate is defined to have the mapping

$$\Sigma a\_x \rightarrow \Sigma(2\mu - a\_x)*x$$

It flips amplitudes around the average, μ. When the "reflection around the mean" occurs with the application of the Grover diffusion operator, almost all the entries x ∈ {0, 1}^n stay roughly the same. But the amplitude of x* gets magnified to about 3/√N in magnitude. Still, this is quite far from our goal of making the amplitude of x* greater than a constant. We achieve this by applying the oracle and the Grover diffusion gate iteratively. The intuitive, high-level picture of what is happening is that a_x* is going up by more than 1/ √ N each time we repeat the application of the pair of gates O(f) and D (Grover diffusion gate). With that logic, after O(√ N) steps, a_x* should exceed the constant 0.1 as desired.

THANK YOU!!