THE UNIVERSITY OF RHODE ISLAND

Quantum Computing

Week 1

Nouman Butt

Institute of AI and Computational Research

January 30, 2026

Introduction

- This is an informal course. No deadlines ☺
- No assigned readings but highly recommended
- No assignments due but would be great if you make an attempt
- Every alternate week we will have a code demo. Again nothing due but play with provided code to sharpen your understanding of the theory and skill development
- This meeting is hybrid(Zoom + in-person). We can start a poll if everyone wants to go fully online.
- Quantum computation uses increasingly sophisticated techniques for heavy-lifting in real computation.
- Goal is to make you comfortable with theory and develop computational skill

Algorithm

A precise recipe for performing a computational task

- What resources are required to perform a given computational task?
- Fundamental model for Algorithms: Turing Machines, simplified form of modern programmble computers
- Quantum Computation is based on ideas in computer science
  - Fourier transform, utilized by many classical algorithms(signal processing), is performed faster in quantum computation
  - Prime Factorization used in RSA public key encryption and RSA digital signature algorithm: An efficient solution is known in Quantum computation namely Shor's algorithm
  - Cryptopgraphic Protocols: Quantum key distribution without any possibility of compromise and quantum random number generators

Classical vs Quantum Bits

- The indivisible unit of classical information is the bit: an object that can take two values: 0 or 1, completely deterministic and stored using transistors, capacitors and other electronic components
- The corresponding unit of quantum information is the qubit, which takes values in a two dimensional complex vector space $\mathcal{C}^2$, naturally probabilistic and built using fundamentals of quantum mechanics

Qubit

$$|\psi\rangle = a|0\rangle + |1\rangle, \quad |a|^2 + |b|^2 = 1$$

where $a, b \in \mathcal{C}$. If we measure $|\psi\rangle$ by projecting it to $\{|0\rangle, |1\rangle\}$, the probability of obtaining $\{|0\rangle, |1\rangle\}$ is $\{|a|^2, |b|^2\}$ respectively.

Computation

- Classical computers with billions of transistors on IC chips perform computational tasks by interpreting and executing instructions(programs) using the ALU for calculations and logical operations, control units for directing operations, registers for data, and cache memory for speed.
- In quantum computers we assemble N qubits in the initial state $\{|0\rangle, |0\rangle, |0\rangle, \ldots |0\rangle\}$. We apply a unitary transformation U, constructed from product of *quantum gates*, to this initial state. Finally, we measure the state by projecting to $\{|0\rangle, |1\rangle\}$. The task performed by the quantum computer is a probabilistic algorithm.

If we run the same quantum program twice we will end up a with a different outcome due to randomness of quantum measurement process.

Classical simulation of quantum process

A classical computer can replicate the quantum algorithm: We say that classical computers can simulate quantum computers to a an arbitrarily good accuracy.

$N = 100$ qubits

Can a classical computer simulate a 100-qubit system with 10^{30} complex numbers to be stored and perform a general rotation in 10^{30} dimensional space.

We can simulate quantum systems of reasonable number of qubits but as N increases the memory print and computational complexity makes the classical simulation inefficient.

Classical vs Quantum State vectors

Consider N classical bits: We have 2^N possible N–bit strings. This configuration space grows exponentially in the number of classical bits. We can store all 2^N strings in separate memory locations/registers.

For a N-qubit quantum system we have a Hilbert(vector) space of dimension 2^N. We have the superposition of all 2^N strings in a generic quantum state and to define this state we need 2^N complex numbers a_x

$$|\psi\rangle = \sum_0^{2^N-1} a_x |x\rangle$$

where $|x\rangle$ represents the number asscoiated with each string in binary notation.

The state of the system is a probablistic mixture of all 2^N classical bit strings.

After measurement we will have a probability distribution over 2^N bit strings. This is not the true distribution that describes the state but rather the distribution we constructed from finite number of samples.

In classical simulations it's the pseudo-distribution we construct by sampling the quantum state via random projections to the computational basis(bitstrings). This is a *classical probalistic algorithm*, in which the outcome is independent of the input,but resultant probability distribution coincides with that generated by the quantum computation.

Quantum mechanics is computationally hard

Evolution of a vector in an exponentially large space

Quantum paralleism

Imagine a function $f(x)$ where x are binary inputs. To evaluate this function classically we will have to iterate through the configuration space(domain).

In quantum computation we prepare a state in the superposition of all states in the configuration space and quantum parallelism allows us to evaluate the function for many different values of x simultaneously.

The difference is that in classical computer evaluations of $f(x)$ on a given inputs exclude one another. In quantum computer the interference of amplitudes allows us to learn some *global* property of the function. Quantum states have a special property called entanglement which is not possible in classical settings.

References

- Quantum Computation and Quantum Information - Nielsen and Chuang
- Quantum Computation John Preskill's notes
- Computational Tools
  - QISKIT
  - https://github.com/PennyLaneAI/pennylane
  - Would recommend using Jupyter notebooks and creating a virtual python environment for installation

THE UNIVERSITY OF RHODE ISLAND

Thank you for your attention

nouman.butt@uri.edu