Notes on Intro to Quantum Computing

Vincenzo Gargano

February 24, 2023

Contents

Chapter 1

Lecture 1

Evolution of quantum technology, at a certain point Google proved that reached a **quantum supremacy**, so can solve intractable problems that other computers can't. What we need to reach quantum supremacy? Find a problem that has a superpolynomial time speedup over the best know possible classic task. Today no machine can implement Shor's algorithm for large number, we need to build a QC that has thousand of qubits and million of gates. In the recent years we had reached very big number of qubits, from 12 in 2006 to 128 in 2020, but more qubits we get more noisy the computation is so we need error correction codes, for correcting errors we need more qubits...

1.0.1 Physical implementation of quantum qubit

Qubit is a very simple system with two basics states, we can use photons with the polarization of a photon, or basic state from nuclear spin in magnetic spin or two state of electron, grounded or excited, or we can shine light on atom (filling with energy), moving electron from one state to another, also we can move electron halfway between this, to us what we matter is to consider them as abstract mathematical objects.

1.0.2 Challenges

From ingeneering point of view we want to reduce interference to make reach ends of operation, also Quantum ops are not perfect, a common operation is to rotate qubit by 90°, but if you end in 90.1° or 89.9° this will introduce an

error that adds up giving incorrect results, also **Incoherence**: losing information due to interaction with environment we lose superposition in short time, Quantum error-correcting codes and fault-tolerant protocols, circuit dimension : suppose to factor a number with n bits shor's algorithm require a ciruit with $\mathrm{O}(\mathrm{n}^2 \log n \log\log n)$.
$Factorizing 2048 - bit number need 150 Millions quantum gates.$

1.1 Circuits: Quantum vs Classical

Architecture are hybrid we have classical computer feeding quantum machine with info and communicating it with quantum circuits. Boolean circuits are non-uniform model of computation, a circuits with n inputs can solve only instances of lenght n.
Non uniform circuits of small size may compute undecidable function, let's consider an undecidable language $L \subseteq \{0,1\}^*$, we can build a circuit that given a string $x \in L$ outputs 1 and 0 otherwise, but we can't build a circuit that given a string $x \in L$ outputs 1 and 0 if x is not in L. We have moved the problem from the language to the two circuit let call them C_n^0 and C_n^1.
Let's impose an uniformity constraint, the family is uniform if each C_n can be constructed by a resource bounded TM, we assume that circuits generated by TM on input n, produce a description of C_n in time polynimial in n and in the number of gates in C_n. So if the circuits contains exponential number of gates will require exponential time as well to produce them.
Measures of complexity are **Size** : Overall number of gates, **Depth** : Longest path from input to output, **Width** : Maximum number of gates active at the same time, there is another called **Space complexity** that is the number of inputs.

1.1.1 Reversible circuits

Quantum circuits have rectangular circuits, so aren't modelled by tree like structures, a computation is reversible (in quantum is resersible always), if it can be performed backwards, there's no loss of information, irreversibility brings to information erasure, reversile computation can recover the input from the output, theory of quantum computing is related to theory of reversible computing. Basically low of physics are reversible.
Reversible circuits require more inputs or outputs and realize a bijections.

Any classical circuit can be made reversile.

Reversible AND gate:
Let x_0, x_1, x_2 be inputs and $x_2 xor(x_0 x_1)$ is the third output.
To make circuit reversible we use a space complexity of O(S+ST)space S and depth T. A set of gates is universal for classical computation for any integers n, m and any function $f : 0, 1^n \to 0, 1^m$ there is a circuit of size $O(nm)$ that computes f.
NAND gate is functionally complete can compute any function, assuming unlimited fan-out assuming that the output can be connected to more logic gates, otherwise we need to add a FANOUT gate that output two copies of a single input (duplicate input). Toffoli gates is universal and can compute all boolen function so we can compute any boolean function on quantum computer, hence the Toffoli gate is a quantum operator.

1.2 Maths of Qubits

How can we describe mathematically a Qubit?
Well using polar notation

$$z = re^{i\phi} = r(\cos\phi + i\sin\phi)) \tag{1.1}$$

Taking the modulo we can see:

$$|e^{i\phi} = |cos\phi + i\sin\phi) = \sqrt{cos^2\phi + sin^2\phi} = 1 \tag{1.2}$$

A qubits is simple a point inside a Bloch sphere, a general description of Qubits is:

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle\, e^{i\theta}\,|\phi\rangle = \cos\theta\,|0\rangle + e^{i\phi}\sin\theta\,|1\rangle \tag{1.3}$$

These two are equivalent, the $e^i\theta$ is global phase factor. We can multiply quibits with this global phases that make no differences, maybe useful to do some algebra. But if a state has a global phase factor in all the both α, β is equivalent to those without global, but if we have $\alpha\,|0\rangle + \beta e^{i\theta}$ is not equivalent.

1.2.1 Bloch sphere

Qubits can be geometrically represented by point on surfaces of spehre of unitary radious, with a representation with 4 degrees of freedom we go in a description with two (can only be in $|1\rangle |0\rangle$)