Quantum Computation and Information

A Gentle Introduction

Pedro Rivero

Argonne National Laboratory
Illinois Institute of Technology
priveroramirez@anl.gov

October 22, 2019

Contents Overview

Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

1 Introduction
- Types of Logic
- Computer Science

2 Quantum Information
- Example: Unitary Transformations
- No-Cloning Theorem

3 Quantum Computation
- Hadamard Gate
- Deutsch's Problem
- Prime Factorization: Shor's Algorithm
- C.T.T. Challenge
- Error Correction

4 Applications
- Quantum Gravity

5 References

What is Quantum Computation?

Introduction
Types of Logic
Computer Science

Quantum Information
Example: Unitary Transformations
No-Cloning Theorem

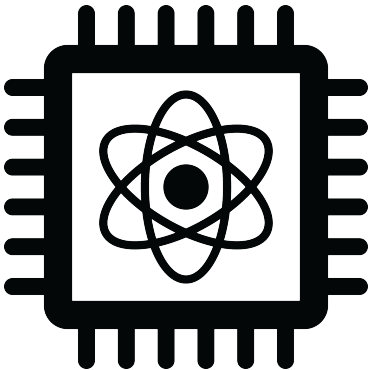
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction

Applications
Quantum Gravity

References

Quantum Computation is the application of **Quantum Information Science (QIS)** to the development of machines capable of performing calculations based on quantum logic instead of the well know classical logic. It is thought to be fundamentally different and much more powerful.

The usefulness of this kind of computation lays not only on the ability to engineer exponentially faster machines, but also on being able to create encryption systems which are fundamentally safe, and simulating nature at the atomic level for game-changing purposes.



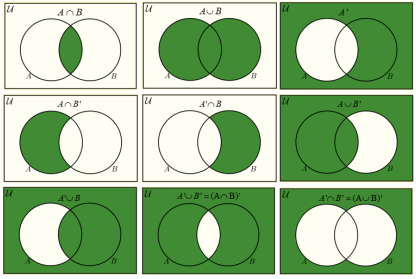
Classical Logic

Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

Classical logic is described by what mathematicians call **Set Theory**, which is easily depicted by the intuitive **Venn Diagrams**. *Phase Space* is indeed the infinite, continuous, multidimensional set of all the possible states of a classical system.

In set theory, the combination of a set with the typical and intuitive logical rules conform what is commonly referred to as a **Boole Algebra**:

- **AND** $\Rightarrow$ $A \cup B$
- **OR** $\Rightarrow$ $A \cap B$
- **NOT** $\Rightarrow$ $\overline{A}$
- **XOR** $\Rightarrow$ $A \cap B - A \cup B$



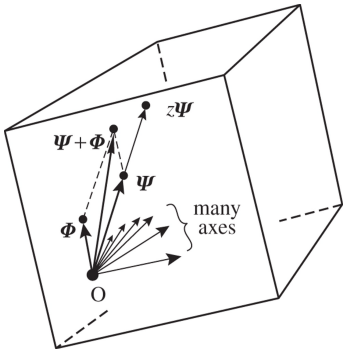
Quantum Logic

Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

Unlike its classical counterpart, quantum logic is described by the mathematical build known as **Vector Spaces**, namely *Hilbert Space*.

Because this logic is in itself different than classical logic, it has some special features:

- **Probabilistic measurement**
- **Measurement causes disturbance**
- **Superposition**
- **Entanglement**
- **Uncertainty principle**

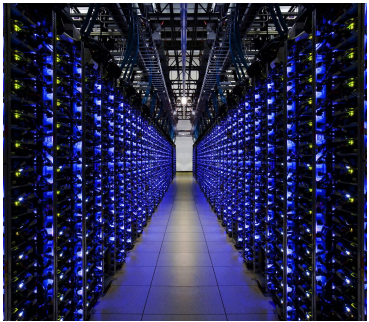


Analog VS Digital

Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

There are several kinds of *"Computers"*. We want to tell apart two major groups:

- **Analog Computers** $\rightarrow$ Solvers for tunable equations
- **Digital Computers** $\rightarrow$ Binary States + Boole Algebra



Church-Turing Thesis

Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

This is **not a theorem**, but rather a **hypothesis** which has been tested and "verified" through experience.

Church-Turing Thesis (CTT)

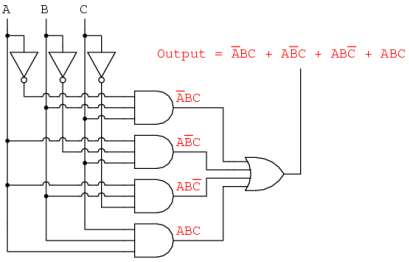
"Ignoring resource limitations such as time or memory, every effective method of computation is either equivalent to or weaker than a Turing machine."

Hence, digital computers are **general purpose** machines that will perform any computation that an analog computer can do, plus they will be able to account for **error correction**.

Algorithms, Circuits and Gates

Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

Computer programs are sequential **algorithms** often expressed as **flow charts**, also known as **circuits**. Along these circuits, the information will be affected by simple logical operations, called **gates**, as to obtain the required final result.



A digital computer will be able to run any sort of algorithm if this one is encoded using a **Turing Complete** programming language – meaning that this language is equivalent to the set of rules used in Turing Machines – as stated by the CTT.

Complexity I

Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

Mathematically, a computer program is simply a mapping of an n-bit input to an m-bit output; this can be expressed as:

$$C_{n,m} : \{0,1\}^n \rightarrow \{0,1\}^m$$

Actually since the output is m separate binary registers assembled together, any computer program is built from m mappings of n-bit inputs to a single bit output:

$$C_n : \{0,1\}^n \rightarrow \{0,1\}$$

Circuit **complexity** is obviously related to the size of the circuit (the number of gates) and the **depth** of the circuit, defined as the number of gates traversed in the longest pathway through the circuit.

Complexity II

Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

Computational complexity is a branch of the theory of computation in theoretical computer science, that focuses on classifying computational problems according to their inherent difficulty, and relating those classes to each other.

- **Class P** are problems which can be solved by a computer in Polynomial time; meaning that the amount of steps necessary to solve a problem scales up as a certain exponent of the number of bits in the input *"n"*. (i.e. Find the biggest number from a given array)
- **Class NP** are problems for which any given solution can be checked in polynomial time. (i.e. Sudokus, Prime factorization...) independently of the time it actually takes to solve them $\Rightarrow$ Good problems for **encryption systems**!

$$P \subseteq NP \quad \Rightarrow \quad P \stackrel{?}{=} NP$$

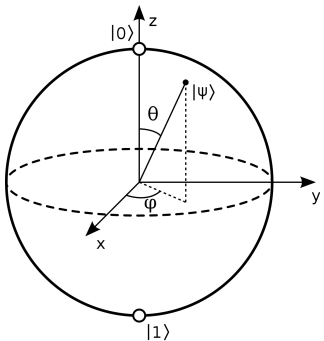
Millenium Prize Problem 1M$

Information Theory

Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

Quantum Information looks at QM from the perspective of trying to determine which are the **constraints** on how nature conserves certain properties through different processes.

The most basic unit of classical-information is called a **bit**. On the other hand, the most basic unit of quantum-information is called a **qubit** and it is nothing but the Hilbert space described by a basis of two distinguishable states.



Example: Unitary Transformations

Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

Example: Preserving information about the distinguishability of two states (Spin-up VS Spin-down) through a transformation, leads to the necessary condition of ***Unitarity****.*

$$\langle 0|1\rangle = 0 \quad (\forall t) \quad \& \quad |\Psi(t)\rangle = U|\Psi\rangle$$

$$\langle 0(t)|1(t)\rangle = \langle 0|U^\dagger U|1\rangle \equiv \langle 0|1\rangle = 0 \quad \Rightarrow$$
$$\Rightarrow U^\dagger U = \mathbb{I} \quad (\mathbb{Q}.\mathbb{E}.\mathbb{D}.)$$

From this **conservation of information** property, it is straight forward to prove that in general $\langle\Phi|\Psi\rangle := \text{const.}(\forall t)$, for any evolution U and any quantum states $|\Psi\rangle$, $|\Phi\rangle$.

No-Cloning Theorem

Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

Classical information can be copied at will. The laws of classical physics allow us to look at some information coded in classical states (i.e. orientation of an arrow) and prepare another classical system so that it perfectly resembles the first.

If we could **clone a quantum state**, then by measuring one of the copies we would be able to learn something about the other without disturbing it. So it has to be impossible in the quantum regime.

Formal Proof I

Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

To prove this we need to think of what a **cloning machine** is. It will be an apparatus with two slots $\{A, B\}$, such that in the beginning the quantum states for each slot are $|\Psi\rangle_A$ (the one we want to clone) and $|e\rangle_B$ a generic initialization state. After the process – which has to be unitary – we will need to get states $|\Psi\rangle_A$ and $|\Psi\rangle_B$ for each slot. In other words:

$$U|\Psi\rangle_A|e\rangle_B = |\Psi\rangle_A|\Psi\rangle_B$$

If we now check what happens for another generic state $|\Phi\rangle$, we will again have:

$$U|\Phi\rangle_A|e\rangle_B = |\Phi\rangle_A|\Phi\rangle_B$$

Formal Proof II

Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

Taking the inner product of these two equations:

$$\langle\Psi|\Phi\rangle = \langle\Psi|_A \langle e|_B |\Phi\rangle_A |e\rangle_B \equiv \langle\Psi|_A \langle e|_B U^\dagger U |\Phi\rangle_A |e\rangle_B$$
$$= \langle\Psi|\Phi\rangle \langle e|U^\dagger U|e\rangle = \langle\Psi|\Phi\rangle \langle\Psi|\Phi\rangle \Rightarrow$$

$$\Rightarrow \langle\Psi|\Phi\rangle = \left(\langle\Psi|\Phi\rangle\right)^2 \Rightarrow \begin{cases} \langle\Psi|\Phi\rangle = 0 \\ \langle\Psi|\Phi\rangle = 1 \end{cases}$$

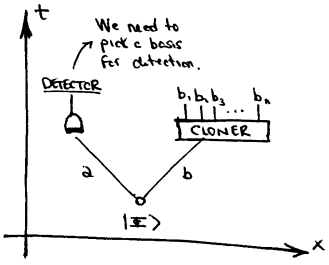
Which means that we can only clone states that form a basis of the corresponding *Hilbert Space* using such unitary transformation U. Therefore, **a single *universal* transformation cannot clone a *general* quantum state**. This is also true for non-unitary transformations.

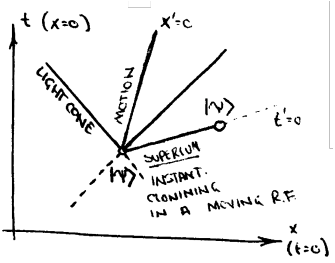
On the other hand, it is straight forward to show in the same manner that **quantum teleportation** is indeed allowed.

Implications

Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

As a matter of fact, there is some deeper reason why this happens. Cloning quantum information will mean **superluminal communication** (faster than light) and viceversa.





- By measuring on different bases we would be able to send a superluminal message.
- In some inertial reference frame, superluminal communication is the same as cloning.

General-Purpose Quantum Computers

Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

For this presentation, we are going to be interested in looking at **digital quantum computers**, how they work and what kind of applications they have. Some examples of qubits used are:

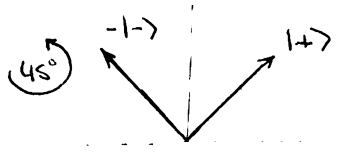
- **Current Loops in superconductors**: $|\circlearrowleft\rangle$, $|\circlearrowright\rangle$
- **Polarization**: $|\updownarrow\rangle$, $|\leftrightarrow\rangle$
- **Spin**: $|\uparrow\rangle$, $|\downarrow\rangle$
- **Energy levels in an atom**: $|\circ\rangle$, $|\odot\rangle$
- **Direction of propagation**: $|\uparrow\rangle$, $|\rightarrow\rangle$

Another thing to notice is that **quantum gates** will only perform unitary transformations, and these transformations will be **reversible**; unlike classical gates.

Hadamard Gate

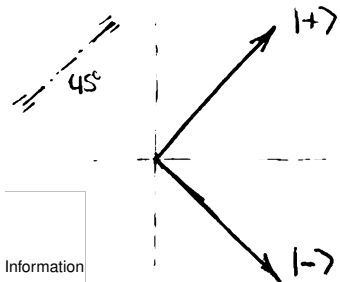
Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

The only, but probably one the most important gates that we want to introduce is the **Hadamard Gate**. This gate performs a 45° rotation followed by a reflection along the 45° axis. Intuitively, we will denote the resulting states $|+\rangle$ and $|-\rangle$.



$$\hat{H} = \underbrace{\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}}_{\text{Reflection}} \underbrace{\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}}_{\text{Rotation}} =$$

$$= \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$



Deutsch's Problem I

Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

Let's try to solve the following problem:

"Given a switch and a bulb, which is the most efficient way (fewer operations) of telling whether they are connected or not?"

Classical:

1. Look at the states of the bulb and the switch once
2. Change the state of the switch
3. Look at the states of the bulb and the switch again
4. Compare 1 and 3

2 MEASUREMENTS

Deutsch's Problem II

Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

Quantum:
We need to express the problem in a more mathematical way in order to tackle it.

$$f : \{0,1\} \to \{0,1\} \quad \Rightarrow \quad \underbrace{f(0) \neq f(1)?}_{\text{Connected}}$$

$$|x\rangle \to \boxed{\textbf{Switch}} \to (-1)^{f(x)} |x\rangle$$

Let's now try to use **superposition** to our advantage by introducing the Hadamard Gate:

$$|0\rangle \to \boxed{\hat{H}} \to |+\rangle \to \boxed{\textbf{Switch}} \to \begin{Bmatrix} \pm|+\rangle \\ \pm|-\rangle \end{Bmatrix} \to \boxed{\hat{H}} \to \begin{Bmatrix} \pm|0\rangle \\ \pm|1\rangle \end{Bmatrix}$$

ONLY ONE MEASUREMENT

Reduction: $2 \to \log_2(2) = 1$

Grandiose Interpretation: Period Finding

Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

It is difficult to see this right away from the example because of the reduced amount of possible states we are considering, but what we are truly doing is **finding the periodicity** of the the function $f(x)$ such that $f(x+T) = f(x)$.

Because we only have two possible states, it is easier to see it using **binary arithmetic**:

$$\begin{cases} 0+0=0 & \text{(even)} \\ 0+1=1+0=1 & \text{(odd)} \\ 1+1=0 \to 10 & \text{(even)} \end{cases}$$

Therefore:

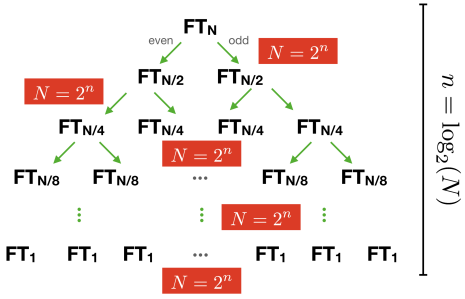
$$\text{If } \left.\begin{cases} f(0) \neq f(1) & \Rightarrow f(x) = f(x+0) \\ f(0) = f(1) & \Rightarrow f(x) = f(x+1) \end{cases}\right\} \Rightarrow \text{Periodicities } \{0,1\}$$

Prime Factorization

Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

Prime factorization is thought to be a problem in the NP class. Checking a given answer is trivial, but performing the task **scales exponentially in time** with the size of the input – at least using the best algorithm we have yet encountered: **Fast Fourier Transform (FFT)**.

- *For any given number M such that* $2^{n-1} \leq M \leq 2^n = N$
- *FFT splits M into its even/odd subproblems*
- *Repeats the same with the subproblems until it is left with N single elements*
- *Recombines*

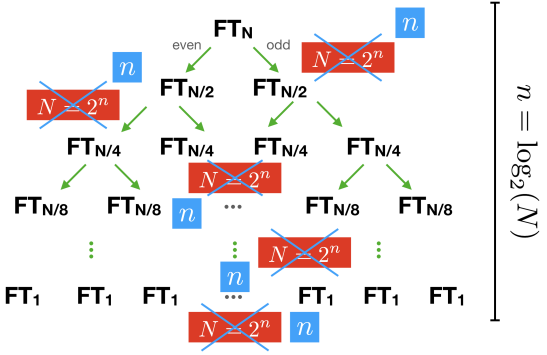


#OPERATIONS: $n \cdot N = n \cdot 2^n$

Shor's Algorithm

Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

MIT Professor **Peter Shor** came up with a quantum algorithm which, similar to what happened in the *Deutsch's Problem*, makes use of superposition to calculate and recombine all the different subproblems in the FFT, reducing the operational cost of each level from $N \to \log_2(N) = n$.



#OPERATIONS: $n \cdot n = n^2$

Prime Factorization: **NP** $\to$ **P**

C.T.T. Challenge

Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

Question: *Does the Church-Turing Thesis extend to Quantum Computers?* In other other words, are all problems solvable by quantum computers also solvable by Turing machines? Given unlimited time and memory.

My current take: YES. As far as one can capture any logical statement on a Turing machine (more or less efficiently), and as far as we have been able to understand quantum mechanics logically (describing them with mathematics) we don't know yet of any problem which a Turing machine cannot compute.

A Quantum Computer can only compute things much more efficiently. Nevertheless, things like entanglement cannot be locally reproduced by a Turing machine **(Bell's Theorem)**, although it can be globally described, and so the question is still unanswered.

Error Correction: Motivations

Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

Because of things like noise, we cannot expect to build qubits in a perfect and ideal way. We will have to account for the possible arising of **errors**.

For this reason we distinguish between:

- **Logical Qubits** ($\# \equiv L$)
- **Physical Qubits** ($\# \equiv P > L$)

This is the main issue that Quantum Computation presents today, as errors spoil the robustness of the more simple computation techniques. Nevertheless, it has been proven that errors $\lesssim 0.01\%$ can be corrected using software and **quantum error-correction techniques** in order to obtain *Full Quantum Computation*. This is called the **Fault-Tolerance Theorem**.

Error Correction: Simple Method

Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

Let's write down some very simple error-correction techniques:

Classical:

$$\text{Substitute: } \begin{cases} \{0\} \to \{0\}\{0\}\{0\} \\ \{1\} \to \{1\}\{1\}\{1\} \end{cases}$$

Quantum: (No-Cloning, measurement disturbs, not discrete...)

$$\text{Entangled } \textbf{qutrits}\text{: } \begin{cases} |0\rangle \to |0,0,0\rangle + |1,1,1\rangle + |2,2,2\rangle \\ |1\rangle \to |0,1,2\rangle + |1,2,0\rangle + |2,0,1\rangle \\ |2\rangle \to |0,2,1\rangle + |1,0,2\rangle + |2,1,0\rangle \end{cases}$$

Any single qutrit in the entangled picture has the exact same $1/3$ probability of being either $\{0,1,2\}$; no matter which original qutrit it corresponds to. Therefore, loosing one will not give out any information and our system will be secured from measurement disturbances. **Information is encoded in the entanglement**.

Applications: Brief List

Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

Some of the foreseeable applications of Quantum Computation which will have a revolutionary impact on society are:

- Cryptography
- Optimization Problems
- Pattern Matching
- Data Analysis
- Deep Learning
- Random Sampling Techniques
- Quantum Simulations
- Chemistry Simulations
- Biology Simulations

Dear Pedro,

Engineers have been talking about quantum computing for decades. But the promise it holds has only gotten close to reality in recent years.

MIT's Quantum Computing online program pairs MIT pedagogy and expert teaching staff with IBM's cutting-edge quantum computing tools, services, and hands-on materials for learners. The courses focus on implementations and applications of quantum computers and include sophisticated labs utilizing IBM Q services.

Still not convinced? Here are 3 reasons why you need to be taking QC seriously:

1. **Cryptography.** Traditional computers make it infeasible to break encryption that uses very large prime number factorization (300+ integers). With QCs, this decryption could become trivial, leading to much stronger protection of our digital lives and assets.
2. **Pattern Matching.** Finding patterns in data and using these to predict future patterns is highly valuable. Consider the possibilities for traffic control, supply chains, or fleet operations and deliveries.
3. **Data Analytics.** Quantum computers will be able to solve complex calculations in mere seconds– the same calculations that traditional computers would take thousands of years to solve. With our rate of data creation growing almost exponentially, this is where the true power of QC will shine!

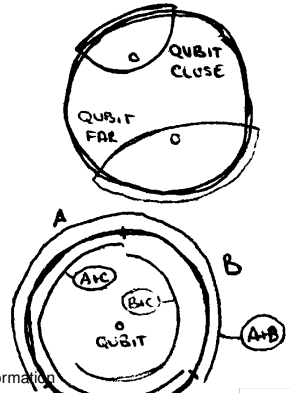
Quantum Gravity

Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

Some important historical facts to know are:

- **Stephen Hawking** $\rightarrow$ Blackhole entropy (information) is proportional to its surface
- **Leonard Susskind** $\rightarrow$ Extended this notion to the Holographic Principle $\Rightarrow$ Q-gravity emerges on the new dim.

The closer the system is to the projecting surface, the smaller the area it will take to encode all of its information.
If it is in the exact center, the information gets distributed as in the quantum error-correcting technique we described, making information about gravity well encoded and protected.



References

Introduction
Types of Logic
Computer Science
Quantum Information
Example: Unitary Transformations
No-Cloning Theorem
Quantum Computation
Hadamard Gate
Deutsch's Problem
Prime Factorization: Shor's Algorithm
C.T.T. Challenge
Error Correction
Applications
Quantum Gravity
References

Hayden, P. *Quantum Computational Universe*

Susskind, L. & Friedman A. *Quantum Mechanics: The Theoretical Minimum*

Nielsen M.A. & Chuang I.L. *Quantum Computation and Quantum Information*

Lykken J. *Quantum Technologies for Quantum Science*

Mermin N.D. *Quantum Computer Science: An Introduction*

Ladd T.D. (et al.) *Quantum Computing*

"The only thing demonstrated by an impossibility proof is a lack of imagination."
– John Stewart Bell –

Argonne
NATIONAL LABORATORY