Quantum Computing

Dr. Andrés Gómez
Andres.gomez.tato@cesga.es
Jan. 2020
Schedule

  - My First Quantum Program.

- Lecture 2: Programming Quantum Algorithms
  - My first Quantum Program with ProjectQ

- Lecture 3: Basic Quantum algorithms

- Lecture 4: Advanced algorithms
Lecture 1

- A brief history of QC and needs.
- Types of quantum computers.
- Basic concepts: qubit, tensors, multiqubit, quantum gates, measurement, amplitudes
- My first quantum program.
- Quantum Circuits. Width, Depth, Quantum Volume.
Welcome to a Dream!

Yuri Manin (1980) and Richard Feynman (1981) proposed independently the concept of Quantum Computer

I’m here very “hot”!!
-273°C

Source: IBM

Welcome to a Dream!

Rigetti

Google

Intel

D-Wave

Qilimanjaro (Spain)

And more in Europe, China, Australia, etc......
Welcome to (my) Nightmare!(*)

Superposition and Entanglement

\[ |\Phi^\pm > = \frac{1}{\sqrt{2}} \left( |00 > \pm |11 > \right) \]

\[ |\Psi^\pm > = \frac{1}{\sqrt{2}} \left( |01 > \pm |10 > \right) \]

(*) When I was a student long time ago!
Quantum Technologies

- Ion Trap
- Transmon
- Quantum Dots
- NV-Defect Diamond
- Photons
- Majorama

And more in the future....
Processor Environment

- Cooled to 0.015 Kelvin, 175x colder than interstellar space
- Shielded to 50,000 × less than Earth’s magnetic field
- In a high vacuum: pressure is 10 billion times lower than atmospheric pressure
- On low vibration floor
- <25 kW total power consumption – for the next few generations
System Shielding

- 16 Layers between the quantum chip and the outside world
- Shielding preserves the quantum calculation
Quantum Computer

- Quantum simulator [1]. Simulate a quantum system using another one, maybe simpler, that can be controlled by the experimenter.

- Adiabatic Quantum Computer [2]. Prepares a known and easy Hamiltonian and lets it evolve to solution.


- Continuous Variable Quantum Computer [5].

- Universal Quantum Computer [3].

$H_B =$ Initial Hamiltonian, which ground state is easy to find

$H_P =$ Problem Hamiltonian, whose ground state encodes the solution to the problem

$H(s) =$ Combined Hamiltonian to evolve slowly:
- $A(s)$ decrease smoothly and monotonically
- $B(s)$ increase smoothly and monotonically

$H(s) = A(s)H_B + B(s)H_P$

A real example: Traffic Flow Optimisation

Preprocess Map & GPS

Find congestions → Find Alternatives → Minimisation model

Redistribute cars

Classical Computer + QPU

D-Wave Adiabatic Computer

Optimisation

Classical Computer + QPU

Unoptimised

Optimised

Xanadu. Continuous Variable

- Language: Strawberry Fields
- Cloud service for Research

https://www.xanadu.ai/

BUILDING A SUPERCOMPUTER ON A SINGLE CHIP
Quantum computing with light can solve in hours or days what would otherwise take billions of years with existing chips.
European Quantum Flagship

- AQTION: Trapped Ions
- OpenSuperQ: Superconducting
- SQUARE: Scalable Rare Earth Ion Quantum Computing Nodes
- MicroQC: Microwave driven ion trap quantum computing

http://qt.eu
Quantum Networks

China Builds World's First Space-ground Integrated Quantum Communication Network

Using Micius for a quantum-safe intercontinental video conference between China and Austria.

The first quantum-safe video conference was held between President BAI Chunli of the Chinese Academy of Sciences in Beijing and President Anton Zeilinger of the Austria Academy of Sciences in Vienna, as the first real-world demonstration of intercontinental quantum communication on September 29th.

Message sending from Vienna to Beijing through space-ground integrated quantum network. (Image by PAN Jianwei’s team)

Private and secure communications are fundamental human needs. In particular, with the exponential growth of Internet use and e-commerce, it is of paramount importance to establish a secure network with global protection of data. Traditional public...
TABLE VI. Runtimes, efficiency and energy consumption for the simulation of random circuit sampling of $N_s$ bitstrings from Sycamore with fidelity $F$ using qFlex on Summit. Simulations used 4550 nodes out of 4608, which represents about 99% of Summit. Single batches of 64 amplitudes were computed on each MPI task using a socket with three GPUs (two sockets per node); given that one of the 9100 MPI tasks acts as master, 9099 batches of amplitudes were computed. For the circuit with 12 cycles, 144/256 paths for these batches were computed in 1.29 hours, which leads to the sampling of about 1M bitstrings with fidelity $F \approx 0.5\%$ (see Ref. [50] for details on the sampling procedure); runtimes and energy consumption for other sample sizes and fidelities are extrapolated linearly in $N_s$ and $F$ from this run. At 14 cycles, 128/524288 paths were computed in 0.72 hours, which leads to the sampling of about 1M bitstrings with fidelity $2.22 \times 10^{-6}$. In this case, one would need to consider 288101 paths on all 9099 batches in order to sample about 1M (0.5M) bitstrings with fidelity $F \approx 0.5\%$ (1.0%). By extrapolation, we estimate that such computations would take 1625 hours (68 days). For $N_s = 3$M bitstrings and $F \approx 1.0\%$, extrapolation gives us an estimated runtime of 1.1 years. Performance is higher for the simulation with 14 cycles, due to higher arithmetic intensity tensor contractions. Power consumption is also larger in this case. Job, MPI, and TAL-SH library initialization and shutdown times, as well as initial and final IO times are not considered in the runtime, but they are in the total energy consumption. *Single precision. **Extrapolated from the simulation with a fractional fidelity.

<table>
<thead>
<tr>
<th>qubits</th>
<th>cycles</th>
<th>$F_{\text{XEB}}$ (%)</th>
<th>$N_s$</th>
<th>nodes</th>
<th>runtime</th>
<th>PFlop/s*</th>
<th>efficiency (%)</th>
<th>power (MW)</th>
<th>energy (MWh)</th>
</tr>
</thead>
<tbody>
<tr>
<td>53</td>
<td>12</td>
<td>0.5</td>
<td>1M</td>
<td></td>
<td>1.29 hours</td>
<td>235.2</td>
<td>111.7</td>
<td>27.3</td>
<td>5.73</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.4</td>
<td>0.5M</td>
<td></td>
<td>1.81 hours**</td>
<td>111.7</td>
<td>57.4</td>
<td>27.3</td>
<td>5.73</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.4</td>
<td>3M</td>
<td></td>
<td>10.8 hours**</td>
<td>111.7</td>
<td>57.4</td>
<td>27.3</td>
<td>5.73</td>
</tr>
<tr>
<td></td>
<td>2.22 $\times 10^{-6}$</td>
<td>1M</td>
<td></td>
<td>4550</td>
<td>0.72 hours</td>
<td>347.5</td>
<td>252.3</td>
<td>61.6</td>
<td>7.25</td>
</tr>
<tr>
<td>14</td>
<td>12</td>
<td>2.22 $\times 10^{-6}$</td>
<td>1M</td>
<td></td>
<td>0.72 hours</td>
<td>347.5</td>
<td>252.3</td>
<td>61.6</td>
<td>7.25</td>
</tr>
<tr>
<td></td>
<td>0.5</td>
<td>1M</td>
<td></td>
<td></td>
<td>67.7 days**</td>
<td>347.5</td>
<td>252.3</td>
<td>61.6</td>
<td>7.25</td>
</tr>
<tr>
<td></td>
<td>1.0</td>
<td>0.5M</td>
<td></td>
<td></td>
<td>67.7 days**</td>
<td>347.5</td>
<td>252.3</td>
<td>61.6</td>
<td>7.25</td>
</tr>
<tr>
<td></td>
<td>1.0</td>
<td>3M</td>
<td></td>
<td></td>
<td>1.11 years**</td>
<td>347.5</td>
<td>252.3</td>
<td>61.6</td>
<td>7.25</td>
</tr>
</tbody>
</table>
Lecture 1

- A brief history of QC and needs.
- Types of quantum computers.
- Basic concepts: qubit, tensors, multiqubit, quantum gates, measurement, amplitudes
- My first quantum program.
- Quantum Circuits. Width, Depth, Quantum Volume.
A scalable physical system with well characterized qubits.

2. The ability to initialize the state of the qubits to a simple fiducial state, such as |000....000>

3. Long relevant decoherence times, much longer than the gate operation time.


5. A qubit-specific measurement capability.

What do you need (today)?

- Complex numbers
- Matrix multiplication
- Understand TENSOR products
- Understand measurement and probabilities
- Imagination
BIT: A “classical” physical system with TWO states

What 0 or 1 means is a convention
Information is codified as a list of BITs
BIT can be transformed from 0 to 1 and vice versa
BITs can be operated with logical gates (OR, XOR, AND…)
One BIT can be cloned
BITs can be stored
BITs can have a long life
BITs move through logical gates
QuBIT: A “Quantum” physical system which yields one of TWO states when is measured

0 \quad \text{OR} \quad 1

What 0 or 1 means is a convention*

Information is codified in several ways
QuBIT can be transformed from 0 to 1 and vice versa
QuBITs can be operated with UNITARY gates
QuBITs cannot be cloned (no-clone theorem)
QuBITs cannot be stored (yet)
QuBITs cannot have a long life (yet)

Usually, QuBITs are quiet
Our current nightmare!

NOISE

https://medium.com/@pchojecki/quantum-advantage-b3458646bd9
Parametric Quantum Circuit Learning

Send Circuit

Return Shots

“QPU”

CPU
Parametric Quantum Circuit Learning

ProjectQ Results

- **sin**
- **exp**
- **par**
- **abs**
Parametric Quantum Circuit Learning

No Noisy backend:qasm_simulator - layout:[1, 0, 3]
Parametric Quantum Circuit Learning
Parametric Quantum Circuit Learning

N=3, D=3

Source: IBM© Nov. 24th, 2019
Parametric Quantum Circuit Learning
Algorithms with shallow circuits

- **QVE**: Quantum Variational Eigensolver: https://arxiv.org/abs/1304.3061


- **Quantum Machine Learning**:
  - Quantum Support Vector Machine
  - Quantum Principal Component Analysis
  - Quantum Variational Autoencoder,
  - Etc.
Quantum Variational Eigensolver

Figure source: Wang, D., Higgott, O., & Brierley, S. (n.d.). A Generalised Variational Quantum Eigensolver.
“Despite a number of promising results, the theoretical evidence presented in the current literature does not yet allow us to conclude that quantum techniques can obtain an exponential advantage in a realistic learning setting”

Ciliberto et.al. “Quantum machine learning: a classical perspective”
http://dx.doi.org/10.1098/rspa.2017.0551

So:

A lot of research to do!!!
If $i^2 = -1$, a complex number is defined by:

$$c = a + b \times i,$$  
with $a, b \in \mathbb{R}, c \in \mathbb{C}$

Complex conjugate: $\bar{c} = a - b \times i$

Modulus: $|c|^2 = c\bar{c} = (a + b \times i)(a - b \times i) = a^2 + b^2$

Polar form: $c = |c| \cos\theta + |c| \sin\theta \times i = |c| e^{i\theta}$
\[ |0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad \text{Superposition} \quad |\phi\rangle = \alpha |0\rangle + \beta |1\rangle \]

\[ |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \]

Amplitude \( \alpha = |\alpha| e^{i\varphi} \)

\[ |\alpha|^2 + |\beta|^2 = 1 \]

\[ |\alpha|^2 \quad 	ext{Probability density} \]

\[ \varphi \quad 	ext{Phase} \]
\[ |\phi> = \alpha |0> + \beta |1> = \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \]
\[ |\psi> = \gamma |0> + \delta |1> = \begin{bmatrix} \gamma \\ \delta \end{bmatrix} \]
\[ <\phi| = \bar{\alpha} <0| + \bar{\beta} <1| = \begin{bmatrix} \bar{\alpha} \\ \bar{\beta} \end{bmatrix} \]
\[ <\phi|\phi> = |\alpha|^2 + |\beta|^2 \]
\[ <\psi|\phi> = \alpha \bar{\gamma} + \beta \bar{\delta} \]

Measurement of \(|\phi>\) in standard basis (\(|0>, |1>\)):

\(|0>\) with probability \(|\alpha|^2\). State after measurement \(|0>\)

or

\(|1>\) with probability \(|\beta|^2\). State after measurement \(|1>\)
Bloch’s Sphere

\[ |\psi > = \cos\left(\frac{\theta}{2}\right) |0 > + \sin\left(\frac{\theta}{2}\right) e^{i\varphi} |1 > \]

\[ \theta = \frac{\pi}{2}, \varphi = 0, |+\rangle = \frac{|0 > + |1 >}{\sqrt{2}} \]

\[ \theta = \frac{\pi}{2}, \varphi = \pi, |-\rangle = \frac{|0 > - |1 >}{\sqrt{2}} \]

\[ \theta = \frac{\pi}{2}, \varphi = \frac{\pi}{2}, |i \rangle = \frac{|0 > + i|1 >}{\sqrt{2}} \]

\[ \theta = \frac{\pi}{2}, \varphi = \frac{3\pi}{2}, |-i \rangle = \frac{|0 > - i|1 >}{\sqrt{2}} \]

Hint: \( e^{i\varphi} = \cos(\varphi) + i \sin(\varphi) \)
One-Qubit Transformations

\[ |\phi > = \alpha |0 > + \beta |1 > = \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \]

- Transform vector space in itself: \( |\phi' > = U |\phi > = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \)
- Unit length vectors must go to unit length vectors: \( \langle \psi | U^\dagger U |\phi > = \langle \psi | \phi \implies U^\dagger U = I \)
- Reversible
- Geometrically, they are rotations of the complex vector space associated to \( |\phi > \)

Note: \( U^\dagger = \bar{U}^T \)
One-Qubit Transformations

- Phase shift: $K(\delta) = e^{i\delta}I$
- Rotation: $R(\beta) = \begin{bmatrix} \cos(\beta) & \sin(\beta) \\ -\sin(\beta) & \cos(\beta) \end{bmatrix}$
- Phase rotation: $T(\alpha) = \begin{bmatrix} e^{i\alpha} & 0 \\ 0 & e^{-i\alpha} \end{bmatrix}$

- Any other Qubit unitary transformation can be written as:

$$K(\delta)T(\alpha)R(\beta)T(\gamma) = \begin{bmatrix} e^{i(\delta+\alpha+\gamma)} \cos(\beta) & e^{i(\delta+\alpha-\gamma)} \sin(\beta) \\ -e^{i(\delta-\alpha+\gamma)} \sin(\beta) & e^{i(\delta-\alpha-\gamma)} \cos(\beta) \end{bmatrix}$$

One-Qubit Transformations

- Phase shift $K(\delta) = e^{i\delta I}$

- Rotation around $x$, $R_x(\theta) \equiv e^{-i\theta x} = \begin{bmatrix} \cos \left(\frac{\theta}{2}\right) & -i \sin \left(\frac{\theta}{2}\right) \\ -i \sin \left(\frac{\theta}{2}\right) & \cos \left(\frac{\theta}{2}\right) \end{bmatrix} = \cos \left(\frac{\theta}{2}\right) I - i \sin \left(\frac{\theta}{2}\right) X$

- Rotation around $y$, $R_y(\theta) \equiv e^{-i\theta y} = \begin{bmatrix} \cos \left(\frac{\theta}{2}\right) & -\sin \left(\frac{\theta}{2}\right) \\ \sin \left(\frac{\theta}{2}\right) & \cos \left(\frac{\theta}{2}\right) \end{bmatrix}$

- Phase rotation, Rotation around $z$, $R_z(\theta) \equiv e^{-i\theta z} = \begin{bmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{bmatrix}$

- Any other QuBit unitary transformation can be written as:

\[ U = K(\delta)R_z(\gamma)R_y(\beta)R_z(\alpha) \]

Source: Nielsen & Chuang, Quantum Computation And Quantum Information
One-Qubit Transformations

- **Pauli Gates**
  - \( X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \), bit-flip or NOT.
  - \( Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \)
  - \( Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \)

- **Clifford group**
  - **Hadamard**, \( H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \)
  - **Phase**, \( S = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix} \)
  - \( \frac{\pi}{8}, T = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix} \)

- **IBM group**
  - \( U_1(\lambda) = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\lambda} \end{bmatrix} \)
  - \( U_2(\phi, \lambda) = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & -e^{i\lambda} \\ e^{i\phi} & e^{i(\phi+\lambda)} \end{bmatrix} \)
  - \( U_3(\theta, \phi, \lambda) = \begin{bmatrix} \cos(\frac{\theta}{2}) & -\sin(\frac{\theta}{2})e^{i\lambda} \\ \sin(\frac{\theta}{2})e^{i\phi} & \cos(\frac{\theta}{2})e^{i(\phi+\lambda)} \end{bmatrix} \)

**Hint:** \( U^\dagger = \overline{U^T} \)
Expectation Value of $U$

$$< U > \equiv < \varphi | U | \varphi >$$

Example:

$$< 0 | Z | 0 > = [1 \ 0] \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} [0] = 1$$

$$< 1 | Z | 1 > = [0 \ 1] \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} [1] = -1$$

$$| \varphi > = a | 0 > + b | 1 >$$

$$< \varphi | Z | \varphi > = (\bar{a} < 0| + \bar{b} < 1|)Z(a| 0 > + b | 1 >) = |a|^2 < 0| Z | 0 > + |b|^2 < 1| Z | 1 >$$
Exercise with 1 QuBit

OPEN QUIRK.HTML
Multi-Qubits

\[ |a\rangle = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} \]

\[ |b\rangle = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \]

\[ \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} \otimes \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} a_1 [b_1] \\ a_1 [b_2] \\ a_2 [b_1] \\ a_2 [b_2] \end{bmatrix} = \begin{bmatrix} a_1 b_1 \\ a_1 b_2 \\ a_2 b_1 \\ a_2 b_2 \end{bmatrix} \]
Multi-Qubits

TENSOR PRODUCT

|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}

|1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}

|0 \otimes |0\rangle = |00\rangle = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = |0\rangle

|1 \otimes |0\rangle = |10\rangle = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} = |2\rangle

|0 \otimes |1\rangle = |01\rangle = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} = |1\rangle

|1 \otimes |1\rangle = |11\rangle = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = |3\rangle
Superposition Multi-Qubits

For 2 QuBits:

$$|\psi\rangle = \alpha|00\rangle + \beta|01\rangle + \gamma|10\rangle + \delta|11\rangle$$

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle + \gamma|2\rangle + \delta|3\rangle$$

For N QuBits:

$$|\psi\rangle = \sum_{i=0}^{2^N-1} \lambda_i |i\rangle$$

Pay Attention. You can map classical information to:

- $|i\rangle$, example Shor’s algorithm and/or
- $\lambda_i$, example HHL algorithm
Entanglement Multi-Qubits

When you cannot write a state as a product of single states

$$|\Phi^{\pm}\rangle = \frac{1}{\sqrt{2}} \left( |00\rangle \pm |11\rangle \right) \neq (\alpha |0\rangle + \beta |1\rangle) \otimes (\gamma |0\rangle + \delta |1\rangle)$$

AND NOW, YOU HAVE WONDERFUL THINGS AS TELEPORTATION!
Let $U_1 = \begin{bmatrix} U_{11} & U_{12} \\ U_{21} & U_{22} \end{bmatrix}$ on qubit 1.

Let $V_2 = \begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix}$ on qubit 2.

$U_1 \otimes V_2 = \begin{bmatrix} U_{11} & V_{11} & U_{12} & V_{12} \\ U_{11} & V_{21} & U_{12} & V_{21} \\ U_{21} & V_{11} & U_{22} & V_{12} \\ U_{21} & V_{21} & U_{22} & V_{22} \end{bmatrix}$

$U_1 \otimes V_2 = \begin{bmatrix} U_{11}V_{11} & U_{11}V_{12} & U_{12}V_{11} & U_{12}V_{12} \\ U_{11}V_{21} & U_{11}V_{22} & U_{12}V_{21} & U_{12}V_{22} \\ U_{21}V_{11} & U_{21}V_{12} & U_{22}V_{11} & U_{22}V_{12} \\ U_{21}V_{21} & U_{21}V_{22} & U_{22}V_{21} & U_{22}V_{22} \end{bmatrix}$
Multi-Qubit Transformations

Example: Apply X gate on second qubit. Let first qubit unchanged

\[ I \otimes X = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 0 & 1 \\ 1 & 0 \\ 1 & 0 \end{bmatrix} \]
Controled Gates

Apply one gate on one qubit, depending on the values of other qubits

\[
\text{CNOT} = |0 \rangle\langle 0| \otimes I + |1 \rangle\langle 1| \otimes X
\]

\[
\text{CNOT} = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 \\
0 & 0 & 1 & 0
\end{bmatrix}
\]

\[
\text{CNOT}|00\rangle = |00\rangle \\
\text{CNOT}|01\rangle = |01\rangle \\
\text{CNOT}|10\rangle = |11\rangle \\
\text{CNOT}|11\rangle = |10\rangle
\]
Measurement

Classical Bit

QuBit
MY FIRST QUANTUM PROGRAM:
Superdense Coding

OPEN QUIRK.HTML
My First Quantum Program

- Using Quirk. Launch quirk.html. **QUIRK does not need measurement. Remember to add it in your real circuit.**

- Apply a Hadamard Gate (H) on the first qubit

- Apply a second H to the same qubit. Result?

- Remove Second H and apply a CNOT on a second qubit.
  - Result: an entangled system (Bell’s)
Superdense Coding

- Transmit two classical bits with a single qubit

A. Bobs generates a Bell’s state
B. Bob sends one qubit to Alice. Bob keeps the second.
C. Alice applies a single-qubit gate to her qubit to encode 2 bits:
   - 01 -> X
   - 10 -> Z
   - 11 -> Y
   - 00 -> I

D. Alice returns her qubit to Bob.
E. Bob uncomputes entanglement (applies the gates in reverse order)
F. Bob measures both qubits.
Superdense Coding

\[ |00\rangle \quad |01\rangle \]

\[ |10\rangle \quad |11\rangle \]
Caution!!!
Exercise 2: IBM Quantum Experience

CONNECT TO: HTTPS://QUANTUM-COMPUTING.IBM.COM/
Quantum Volume

- **Width**: The number of physical qubits;
- **Depth**: The number of gates that can be applied before errors make the device behave essentially classically;
- **Topology**: The connectivity of the device;
- **Gate Parallelism**: The number of operations that can be run in parallel.
TOPOLOGY


https://github.com/Qiskit/ibmq-device-information/blob/master/backends/melbourne
Quantum Volume

- Effective error rate $\varepsilon_{\text{eff}}$: specifying how well a device can implement arbitrary pairwise interactions between qubits

- $n$ is the number of qubits of the Computer

- $n'$ number of qubits used by the algorithm

- Depth $d \approx \frac{1}{n \varepsilon_{\text{eff}}}$

- Quantum Volume

$$V_Q = \max_{n'<n} \min \left[ n', \frac{1}{n' \varepsilon_{\text{eff}}(n')} \right]^2$$

Source: IBM, 2019
## Classical Resources

<table>
<thead>
<tr>
<th>1 qubit</th>
<th>2 qubits</th>
<th>3 qubits</th>
<th>N qubits</th>
</tr>
</thead>
<tbody>
<tr>
<td>$</td>
<td>0\rangle$</td>
<td>$</td>
<td>00\rangle =</td>
</tr>
<tr>
<td>$</td>
<td>1\rangle$</td>
<td>$</td>
<td>01\rangle =</td>
</tr>
<tr>
<td>$</td>
<td>10\rangle =</td>
<td>2\rangle$</td>
<td>$</td>
</tr>
<tr>
<td>$</td>
<td>11\rangle =</td>
<td>3\rangle$</td>
<td>$</td>
</tr>
<tr>
<td>$</td>
<td>110\rangle =</td>
<td>6\rangle$</td>
<td>$</td>
</tr>
</tbody>
</table>

$2 \cdot \text{complex}= 2 \times 2 \times 8 = 32 \text{ bytes}$

$4 \times 2 \times 8 = 64 \text{ bytes}$

$8 \times 2 \times 8 = 128 \text{ bytes}$

$2^N \times 2^4 = 2^{N+4}$
Asimov calculated the number of nucleons+electrons in the Universe as $\sim 10^{79}$

$\sim 10,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000$

Having a QPU with 270 qubits, one can store in the amplitudes: $\sim 10^{81}$ FPs.

Year 2025: $\sim 170\text{ZB/year} \sim 10^{23} \text{bytes/year}$

75 qubits: $\sim 3 \cdot 10^{23} \text{ FPs} = \sim 24 \text{ years}!!!
## Classical Resources

<table>
<thead>
<tr>
<th>qubits</th>
<th>RAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>32 bytes + memory for gates</td>
</tr>
<tr>
<td>2</td>
<td>64 bytes + memory for gates</td>
</tr>
<tr>
<td>3</td>
<td>128 bytes + memory for gates</td>
</tr>
<tr>
<td>4</td>
<td>256 bytes + memory for gates</td>
</tr>
<tr>
<td>8</td>
<td>4 kbytes + memory for gates</td>
</tr>
<tr>
<td>16</td>
<td>1 Mbytes + memory for gates</td>
</tr>
<tr>
<td>32</td>
<td>64 Gbytes + memory for gates</td>
</tr>
<tr>
<td>36</td>
<td>1TB + .....</td>
</tr>
<tr>
<td><strong>38</strong></td>
<td><strong>4TB (Limit CESGA FT2 FAT node ....)</strong></td>
</tr>
<tr>
<td>45</td>
<td>0.5PB [1]</td>
</tr>
<tr>
<td>64</td>
<td>512 ExaBytes!!!</td>
</tr>
</tbody>
</table>


**THIS IS ONLY TRUE IF YOU NEED ALL POSSIBLE STATES!**
End of Lecture 1
Questions?