Quantum Computing

Lecture 2

Ahmed Abdelmotteleb

University of Warwick

ahmed.abdelmotteleb@cern.ch

CERN openlab Summer Lectures

26th of July 2023

“The history of the universe is, in effect, a huge and ongoing quantum computation. The universe is a quantum computer.”

-Seth Lloyd

Mathematical aside 2 – Matrix Operations

- Quantum theory is **unitary**, a unitary matrix U is such that $\boxed{U^\dagger U = I_n,}$ where I_n is the identity matrix and n represents the dimension of the square matrix U

$\dagger$ = Hermitian conjugate

$$\boxed{e.g.\, I_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}}$$

- $U^\dagger$ (U-dagger) is the transposed, complex-conjugated version of the matrix U

- Let $U = \begin{pmatrix} U_{00} & U_{01} \\ U_{10} & U_{11} \end{pmatrix}$, $\quad U^\dagger = \begin{pmatrix} U_{00}^* & U_{10}^* \\ U_{01}^* & U_{11}^* \end{pmatrix}$ $\quad (a = 2 - 3i \quad \rightarrow \quad a^* = 2 + 3i)$

- Can express U in Dirac notation as follows:

$$U = \begin{pmatrix} U_{00} & U_{01} \\ U_{10} & U_{11} \end{pmatrix} = U_{00}|0\rangle\langle 0| + U_{01}|0\rangle\langle 1| + U_{10}|1\rangle\langle 0| + U_{11}|1\rangle\langle 1|$$

e.g. $U = \begin{pmatrix} 5 & 6 \\ 7 & 8 \end{pmatrix} = 5|0\rangle\langle 0| + 6|0\rangle\langle 1| + 7|1\rangle\langle 0| + 8|1\rangle\langle 1|$

Matrix Operations - continued

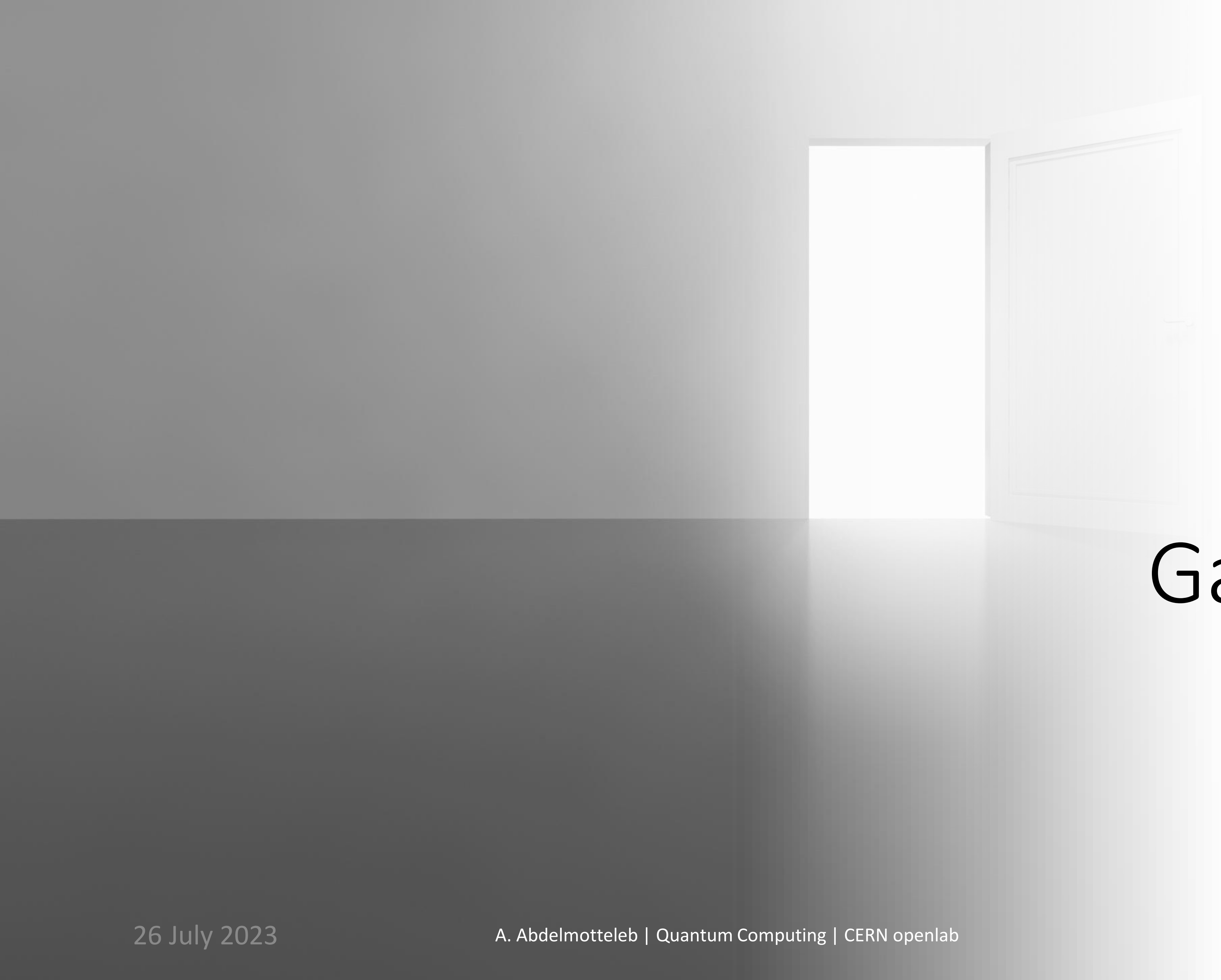
- Recap on how to multiply two 2x2 matrix with a 2x1 matrix (2D-vector):

- A = $\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$ B=$\begin{pmatrix} 5 \\ 6 \end{pmatrix}$

- AB = $\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}\begin{pmatrix} 5 \\ 6 \end{pmatrix}$

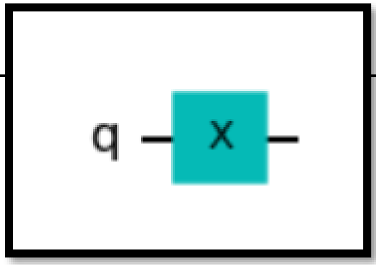
$$= \begin{pmatrix} 1.5 + 2.6 \\ 3.5 + 4.6 \end{pmatrix}$$

$$= \begin{pmatrix} 17 \\ 39 \end{pmatrix}$$

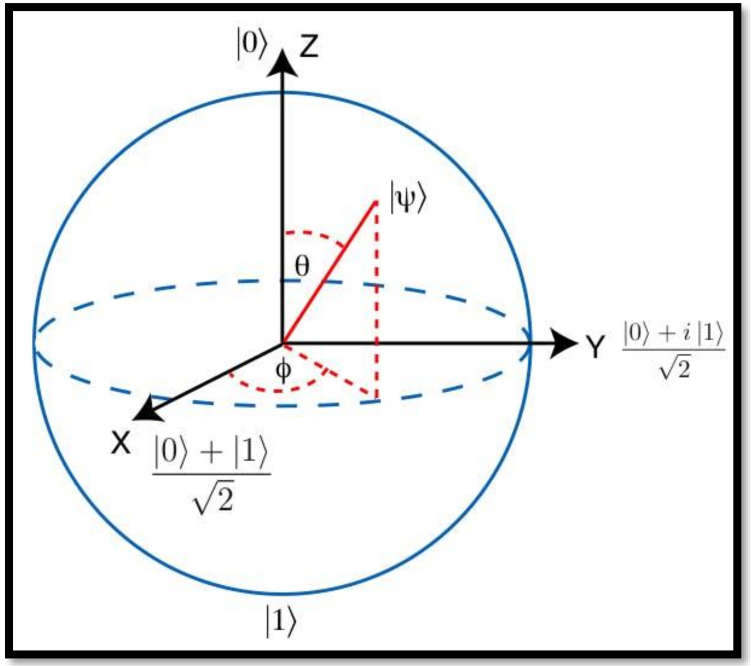


Gates

Quantum Logic Gates (qubit gates)



- Quantum computing relies on **quantum circuits**
- A quantum circuit is a sequence of blocks or gates that carry out computations (input-output)
- A quantum gate is represented by unitary matrices ($U^{\dagger}U = I_n$)
- **Pauli (spin) matrices (gates):** $\sigma_x, \sigma_y, \sigma_z$
- $\boxed{\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = |0\rangle\langle 1| + |1\rangle\langle 0|}$



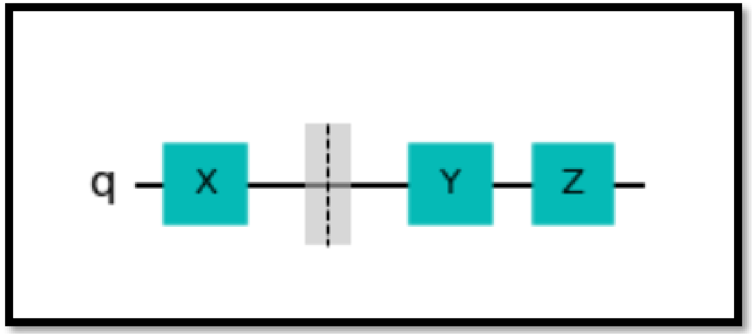
  - $\sigma_x|0\rangle = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} = |1\rangle$
  - $\sigma_x|1\rangle = (|0\rangle\langle 1 + |1\rangle\langle 0|).|1\rangle = |0\rangle\overbrace{\langle 1|1\rangle}^{1} + |1\rangle\cancelto{0}{\langle 0|1\rangle} = |0\rangle$

Bit flip (rotation about x-axis by π) (analogous to classic NOT gate)

Quantum Logic Gates (qubit gates) - continued

- $\boxed{\sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} = |0\rangle\langle 0| - |1\rangle\langle 1|}$

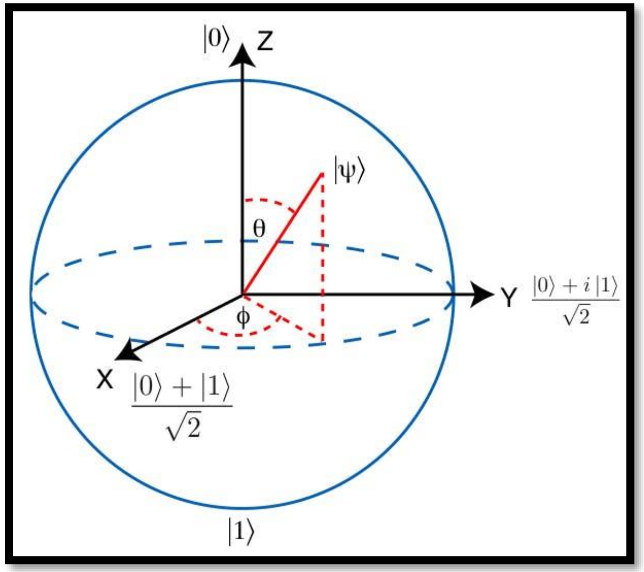
  - $\sigma_z|+\rangle = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix} = |-\rangle \quad (|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle))$
  - $\sigma_z|-\rangle = (|0\rangle\langle 0| - |1\rangle\langle 1|).\left(\frac{1}{\sqrt{2}}|0\rangle - |1\rangle\right) = \frac{1}{\sqrt{2}}|0\rangle + |1\rangle = |+\rangle$



Phase flip (rotation about z-axis by π)

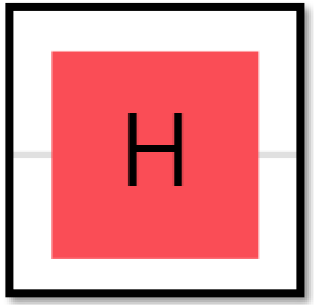
- $\boxed{\sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} = -i|0\rangle\langle 1| + i|1\rangle\langle 0| = i\sigma_x.\sigma_z}$

Bit and phase flip (rotation about y-axis by π)



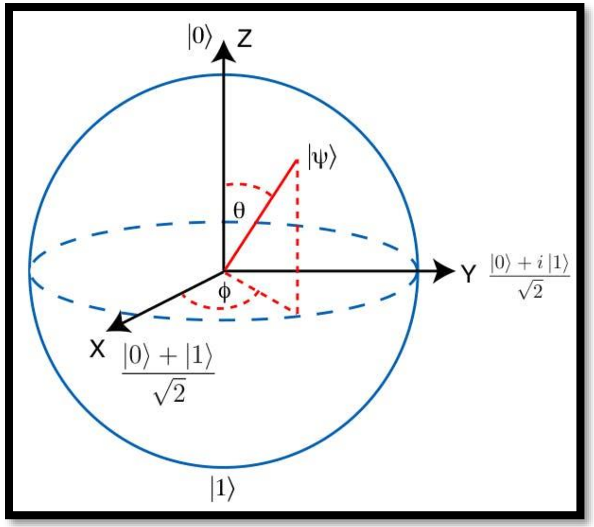
- $\sigma_i^2 = I_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$, i.e. applying the same Pauli gate twice does nothing to a state

Quantum Logic Gates (qubit gates) - continued



Hadamard gate:

- Can be found in (almost) every quantum circuit
- $\boxed{H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = \frac{1}{\sqrt{2}}(|0\rangle\langle 0| + |0\rangle\langle 1| + |1\rangle\langle 0| - |1\rangle\langle 1|)}$



➢ $H|0\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 1 \end{pmatrix} = |+\rangle$

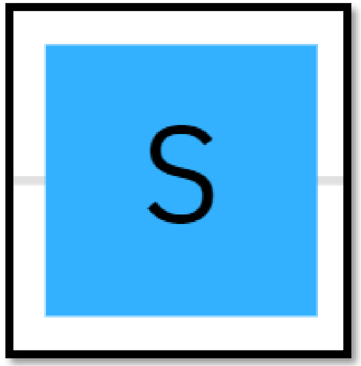
➢ $H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle\langle 0| + |0\rangle\langle 1| + |1\rangle\langle 0| - |1\rangle\langle 1|).\,|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = |-\rangle$

➢ $H|+\rangle = |0\rangle$ $\qquad$ $H|-\rangle = |1\rangle$

➡ **Creates and destroys superposition (switch between z and x bases)**

Quantum Logic Gates (qubit gates) - continued

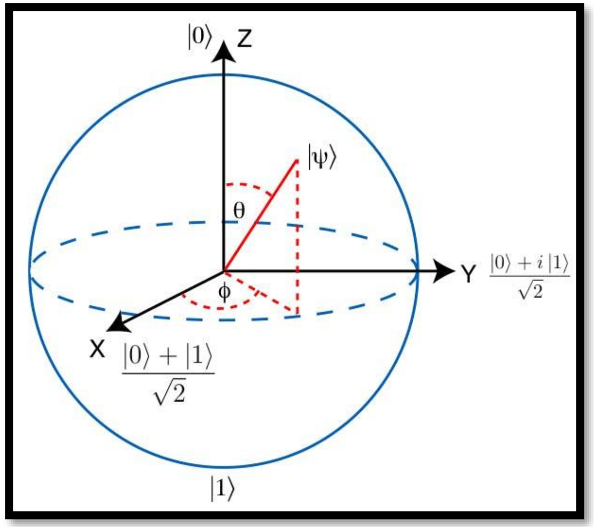
S gate:



- $$\boxed{S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} = |0\rangle\langle 0| + i|1\rangle\langle 1|}$$

  - $S|+\rangle = |i\rangle$ $\quad$ $S|-\rangle = |-i\rangle$

➡ **Adds $\frac{\pi}{2}$ (90°) to the phase φ (switch between x and y bases)**



- $\boldsymbol{S.H}$ allows us to **switch between z and y bases**

https://javafxpert.github.io/grok-bloch/

Two qubits and tensor products

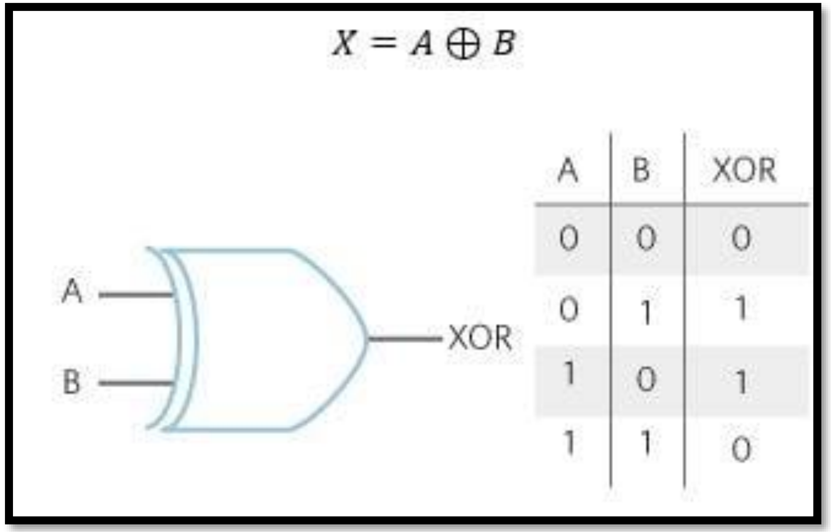
$$|00\rangle = \begin{pmatrix}1\\0\\0\\0\end{pmatrix} \qquad |01\rangle = \begin{pmatrix}0\\1\\0\\0\end{pmatrix} \qquad |10\rangle = \begin{pmatrix}0\\0\\1\\0\end{pmatrix} \qquad |11\rangle = \begin{pmatrix}0\\0\\0\\1\end{pmatrix}$$

- $\langle e|d\rangle = \langle ed\rangle -$ inner product
- $|d\rangle\langle e| = |de| -$ outer product
- $|d\rangle|e\rangle = |d\rangle\otimes|e\rangle = |de\rangle -$ tensor product

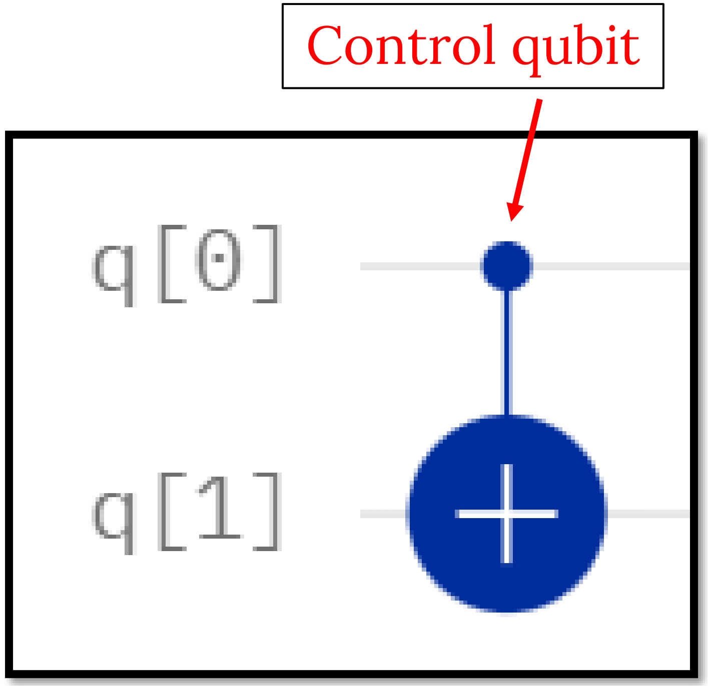
e.g. $|0\rangle_1|0\rangle_2 = |0\rangle_1\otimes|0\rangle_2 = |0_1 0_2\rangle$

Two qubit gates

CNOT gate (CX):



Input		Output	
x	y	x	x⊕y
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0



- Note that a classical XOR gate is non-reversible
- However, since all quantum gates are unitary, they are, in fact, **reversible**
- CNOT stands for Controlled-NOT. The qubit in control does not change by the gate

Two qubit gates - continued

CNOT gate:

- $$CNOT = \begin{array}{c c} & \begin{array}{cccc} 00 & 01 & 10 & 11 \end{array} \\ \begin{array}{c} 00 \\ 01 \\ 10 \\ 11 \end{array} & \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \end{array} = |00\rangle\langle 00| + |01\rangle\langle 01| + |10\rangle\langle 11| + |11\rangle\langle 10|$$

  - $$CNOT|00\rangle = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} = |00\rangle$$

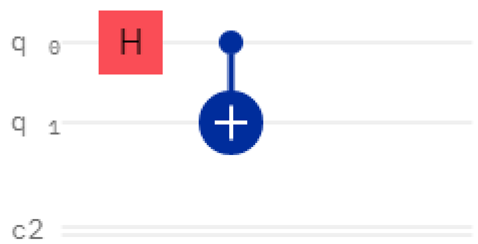
  - $$CNOT|10\rangle = (|00\rangle\langle 00| + |01\rangle\langle 01| + |10\rangle\langle 11| + |11\rangle\langle 10|).|10\rangle$$

$$|00\rangle\langle 00|10\rangle + |01\rangle\langle 01|10\rangle + |10\rangle\langle 11|10\rangle + |11\rangle\langle 10|10\rangle$$
$$= |11\rangle$$

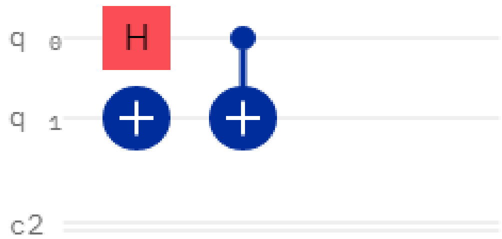
Bell states

- 4 states (2 qubits) that are **maximally-entangled** and build an **orthonormal basis**

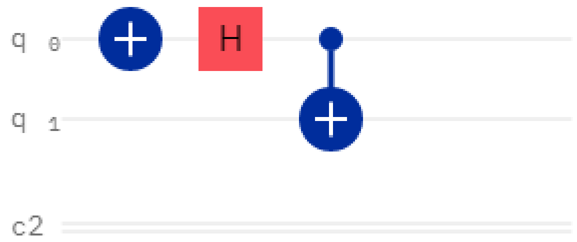
$$|\psi^{00}\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$



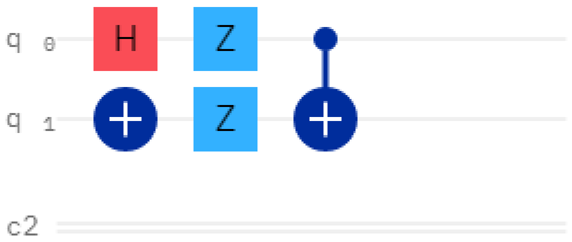
$$|\psi^{01}\rangle = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle)$$



$$|\psi^{10}\rangle = \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle)$$



$$|\psi^{11}\rangle = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle)$$



More entanglement

$$\boxed{|\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)} \longrightarrow \text{Global state}$$

$$|\psi_1\rangle = \alpha_1|0\rangle + \beta_1|1\rangle \longleftarrow \text{local state} \longrightarrow |\psi_2\rangle = \alpha_2|0\rangle + \beta_2|1\rangle$$

Given the global state, what are the local states?

$$|\psi_1\rangle \otimes |\psi_2\rangle = |\psi_1\psi_2\rangle = \boxed{\alpha_1\alpha_2|00\rangle} + \boxed{\alpha_1\beta_2|01\rangle + \beta_1\alpha_2|10\rangle} + \boxed{\beta_1\beta_2|11\rangle}$$

Therefore, require:

$\otimes$ – tensor product

- $\alpha_1\beta_2 = \beta_1\alpha_2 = 0$
- $\alpha_1\alpha_2 = \beta_1\beta_2 = \frac{1}{\sqrt{2}}$

$\Rightarrow$ $\alpha_1 = 0$ or $\beta_2 = 0$

Contradiction!

More entanglement - continued

$$|\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |10\rangle) = \frac{1}{\sqrt{2}}(|0\rangle \otimes |0\rangle + |1\rangle \otimes |0\rangle) = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |0\rangle$$

$$|\psi\rangle = \frac{1}{\sqrt{2}}(|10\rangle + |01\rangle) = \frac{1}{\sqrt{2}}(|1\rangle \otimes |0\rangle + |0\rangle \otimes |1\rangle) = ?$$

We have full knowledge of the global state, but no knowledge of the local states.

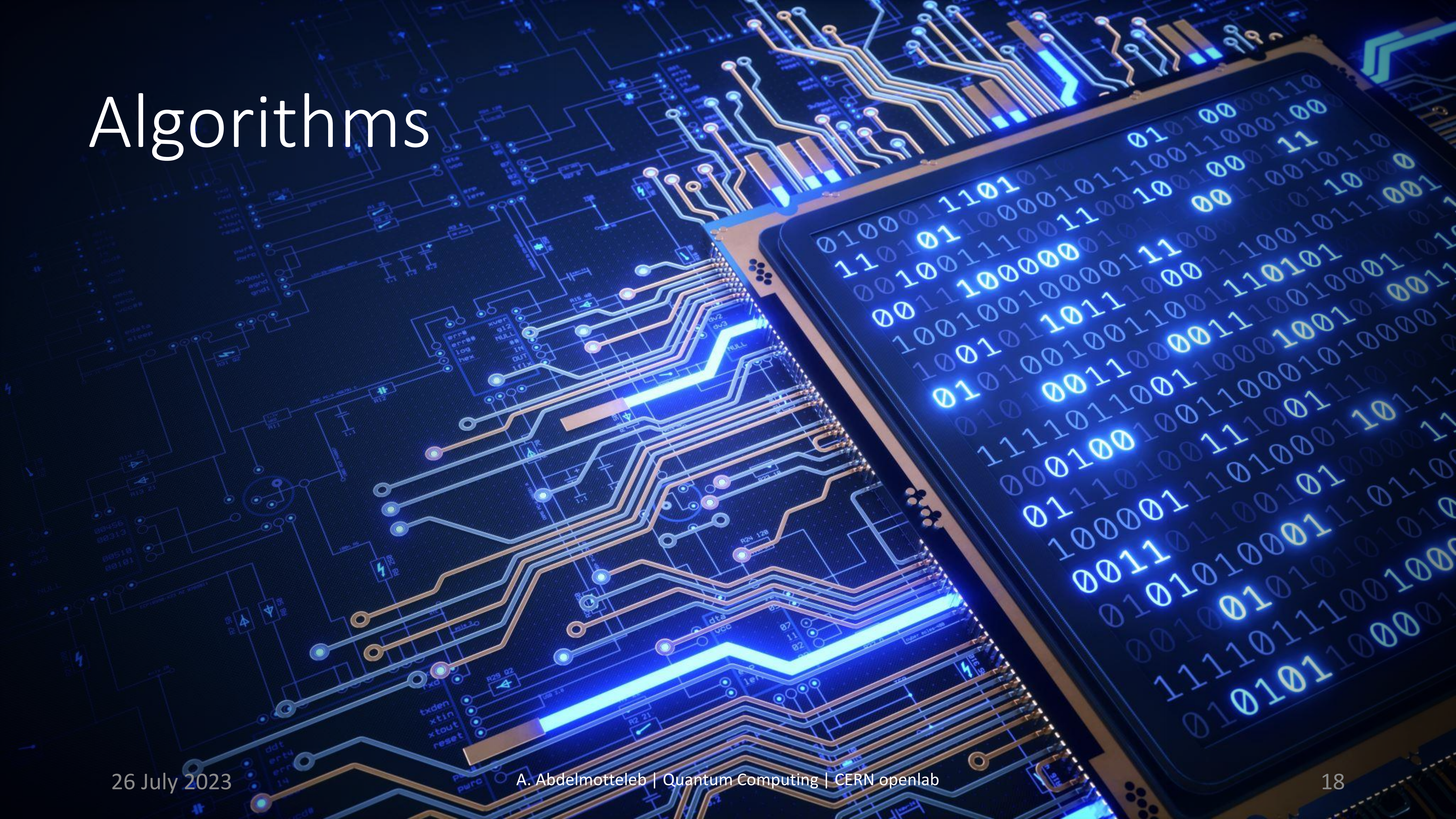
Entanglement!

Gates summary

- Pauli (spin) matrices (gates) cause spins around x,y, or z axes of a Bloch sphere (bit or phase flip)
- Hadamard gates create and destroy superposition
- S gate(also in combination with H gate) switches between bases
- CNOT gates (2 qubit gate) help us achieve entanglement
- Bell states (4 of them – 2 qubits) are maximally-entangled and build an orthonormal bases
  → help you get a Nobel prize

Operator	Gate(s)		Matrix
Pauli-X (X)	$\boxed{X}$	$\oplus$	$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
Pauli-Y (Y)	$\boxed{Y}$		$\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$
Pauli-Z (Z)	$\boxed{Z}$		$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$
Hadamard (H)	$\boxed{H}$		$\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$
Phase (S, P)	$\boxed{S}$		$\begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}$
$\pi/8$ (T)	$\boxed{T}$		$\begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix}$
Controlled Not (CNOT, CX)	$\bullet$ / $\oplus$		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$

Algorithms



Grover's Algorithm

- An (quantum) algorithm used to search and locate a specific element in an unordered list/unsorted database
- Imagine you have a list of 2 columns: name and phone number
- You have a phone number, and you want to find the corresponding name using this list
- Using a classical computer, you would need to use brute-forcing or some other method (not highly efficient)
- With Grover's algorithm and the superposition principle, one can exponentially decrease the time needed to find the phone number
- This process happens with the help of two sub-functions called the **oracle function** and the **amplification function**
- Other fun examples is solving a **sudoku puzzle** or **polynomial root** finding problems

Grover's Algorithm - continued

- Suppose you have 2 qubits, corresponding to 4 possibilities: 00, 01, 10, and 11

- All possible states can be described using this equation:

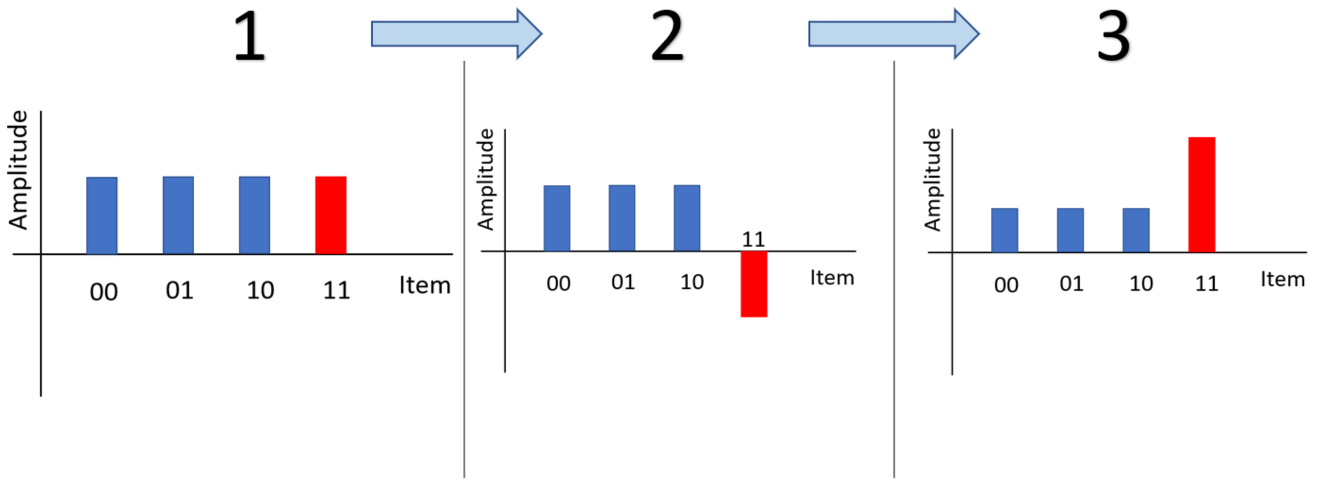
$$a|00\rangle + b|01\rangle + c|10\rangle + d|11\rangle,$$

where a, b, c, and d correspond to the amplitudes of the states (remember, probability of the state is the amplitude-squared)

- Assume for example that our phone number is associated with the state $|11\rangle$

- This means the associated amplitude of interest is "d"

Grover's Algorithm - continued



- The oracle function **flips** the corresponding amplitude "d" so that it becomes "$-d$"; a **unique amplitude**
- The amplification function **amplifies** the difference between the amplitude corresponding to the number we're looking and the other amplitudes
- This makes the probability of locating the number much higher
- **Repeat** as many times as one pleases to further increase the probability

Grover's Algorithm - demo

More details/explanations can be found here (Qiskit Textbook)

Cryptography

- Protecting information and communications using codes
- RSA cryptography utilizes **prime numbers** to securely encrypt data. It is fundamental for the operation of internet protocols
- Need to have the factors of a number to be able to decrypt data
- E.g. given the number a number $n = pq = 226{,}579$, try to find p, q, given that they are prime numbers (answer: $p = 419$, $q = 541$)
- Nowadays, we use **RSA-2048** which utilized a 2048 bit key (an integer on the order of 2^{2048})
- Estimated it would take a classical computer around **300 trillion years** to break an RSA-2048-bit encryption key

Example of an RSA-2048 integer

25195908475657893494027183240048398571429282126204032027777137836043662020707
595556264018525880784406918290641249515082189298559149176184502808489120072
8449926873928072877767359714183472702618963750149718246911650776133798590957
0009733045974880842840179742910064245869181719511874612151517265463228221686
998754918242243363725908514186546204357679842338718477444792073993423658482
382428119816381501067481045166037730605620161967625613384414360383390441495
263443219011465754445417842402092461651572335077870774981712577246796292638635
6373289912154831438167899885040445364023527381951378636564391212010397122822
120720357


RSA Cryptosystem: *exists*
Quantum Computers:

(very basic) introduction to Shor’s Algorithm

- By no means a proper explanation to how Shor’s Algorithm works – actual explanation could take more than one lecture

- Using a perfect quantum computer, finding the factors of an RSA-2048-bit integer could take mere **seconds**

- The basic premise is that you setup a **special periodic function** (modulo function), with a period “r”, which you try to obtain. This then leads you to obtaining the factors of the big number

- With **superposition**, you can test many values for the period simultaneously

- You use **interference** to zero-in on the correct value of the period (constructive interference) and reduce the probability of landing on the incorrect value (destructive interference)

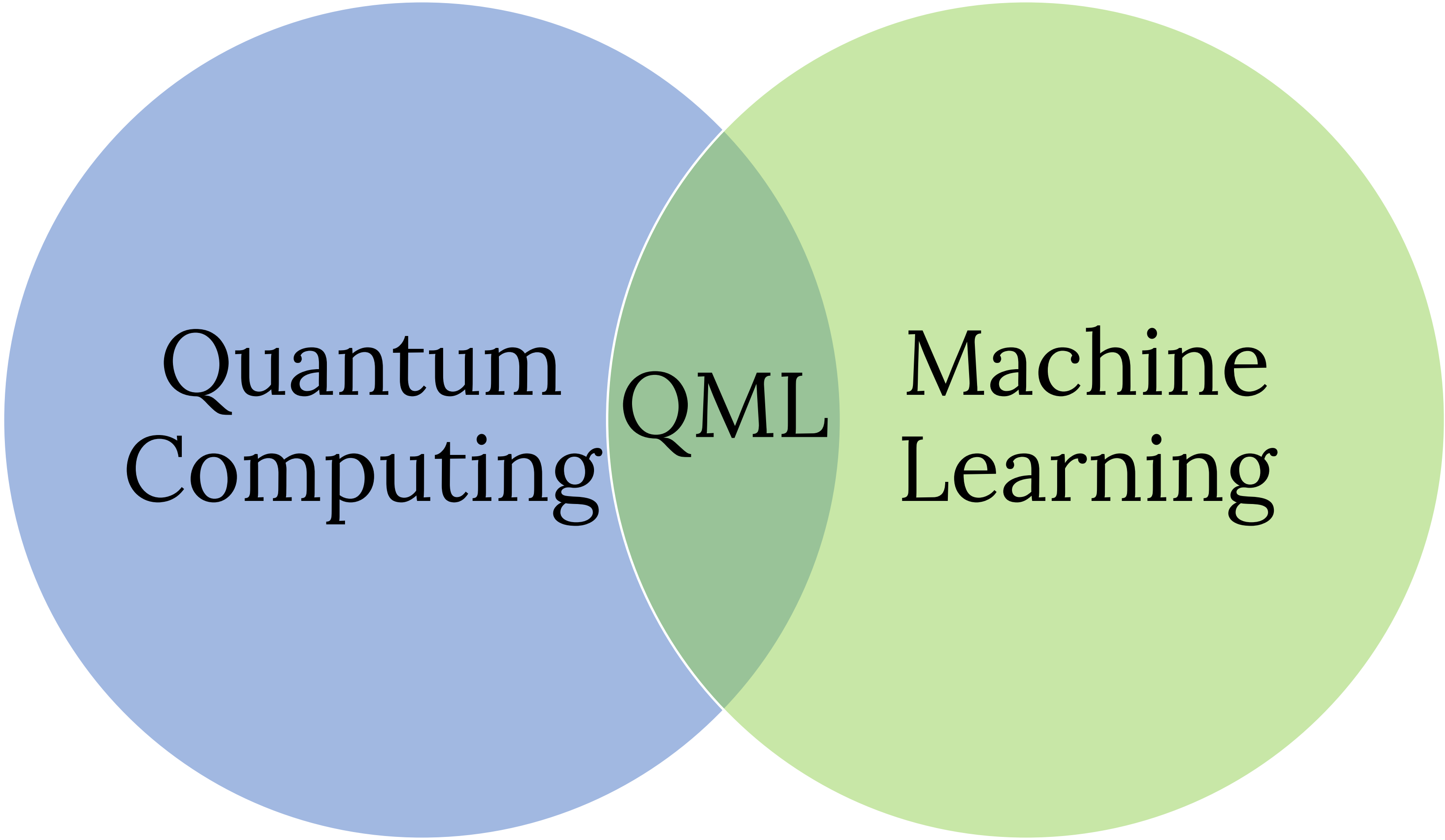
- To find the period of this modulo function, you use **“Quantum Fourier Transform”** – a very useful tool in quantum computing

Quantum-safe cryptography

- RSA encryption is the basis of a lot of encryption schemes used to protect many assets
- Shor's algorithm poses a threat to these kind of commonly used encryption methods
- Need to have a quantum computer with **millions** of physical qubits to be able to do that (due to decoherence and noise)
- Quantum-safe algorithms which typically rely on mathematical problems that can't be solved easily by both classical and quantum computers already available
- Algorithms that rely on geometric problems based on lattices such as CRYSTALS-Kyber and CRYSTALS-Dilithium are now recommended by NIST
- Expected that industries will transition over to quantum-safe cryptographic algorithms as soon as **2024**. Therefore, there is (almost) no reason to worry about quantum computers destroying the world

Current applications of quantum computing

Quantum Machine Learning (QML)



Quantum Machine Learning (QML) - continued

- Machine Learning is linear algebra, and quantum computing heavily relies on linear algebra → excellent match up

- QML can be used to solve Fourier Transformation, finding eigenvectors and eigenvalues, and solving linear sets of equations with an **exponential speedup**

- **Quantum Support Vector Machines** (QSVM) are also one of the more popular QML techniques.
  - Classical SVMs can be performed only up to a certain number of dimensions while QSVMs do not suffer from these restrictions

- **Quantum Optimization:** try to produce best possible output by using the least possible resources.
  - Superposition, entanglement and interference

Current applications of quantum computing

- **Condensed matter physics** has important implications for our understanding of nature and the development of new technologies
- It is also one of the main building blocks behind building computers, both classical and quantum
- One of the most popular models of ferromagnetism in statistical mechanics is called the "**Ising model**"
- Utilizes spins as its variables, and its coefficients comprise couplings and fields
- Each spin configuration is assigned an energy and a corresponding Boltzmann probability
- The connectivity of qubits allows researchers to simulate the dynamics of spin lattices

Current applications of quantum computing - continued

- One of many research papers on this subject include this one from [IBM Quantum]
- They perform Markov chain Monte Carlo (MCMC), a popular iterative sampling technique, to sample from the Boltzmann distribution of classical Ising models
- A new quantum algorithm that with many current applications including ones in machine learning (Boltzmann Machines) and statistical physics (thermal averages)
- Uses relatively simple quantum circuits using current hardware
- Many similar efforts trying to create solutions for optimization, statistical, and machine learning problems

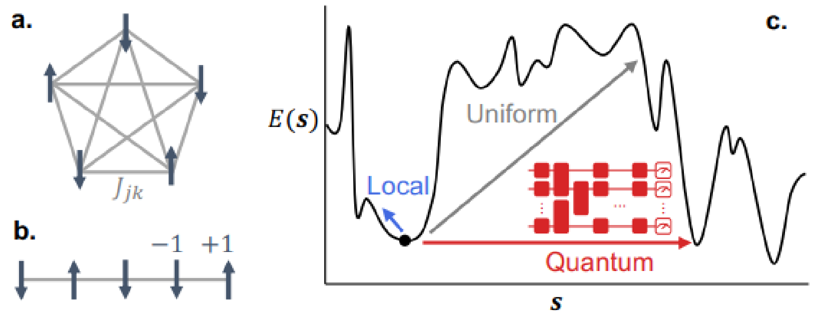


FIG. 1. **Ising model representations. a.** Graph depicting an $n = 5$ model instance where arrows (vertices) represent spins and edges represent the $\binom{n}{2}$ non-zero couplings J_{jk}. Fields h_j are not shown. **b.** An $n = 5$ model instance with only $n - 1$ non-zero couplings. **c.** A rugged energy landscape typical of spin glasses, with the configurations $\boldsymbol{s} \in \{-1, 1\}^n$ depicted in 1D. Typical proposed jumps for three MCMC algorithms, from a local minimum, are shown for illustration.

[arXiv:2203.12497]

Current applications of quantum computing - HEP

- Many applications in HEP, focusing on things like track reconstruction, Lattice Electrodynamics, signal versus background separation, and finding rare processes with QML

- The field of quantum computing in HEP is super recent → lots of opportunities to explore!

- I am going to be biased and focus on one recent paper published by my current experiment, LHCb

- Main premise is utilizing QML to identify the **charge of the b hadron-jets**

- Proponents utilize a Variational Quantum Classifier (VQC) on LHCb data and compare it against a Deep Neural Network (DNN) model

Current applications of quantum computing – HEP – continued

- Measurements mapped to probabilities for different labels
- Probabilities used to estimate a cost function
- Cost function optimized using a classical optimizer
- QML algorithms achieve performance consistent with classical methods like the DNN with low-complexity circuits and a smaller number of training events
- Room for improvement, especially when using a larger number of features and utilizing better hardware

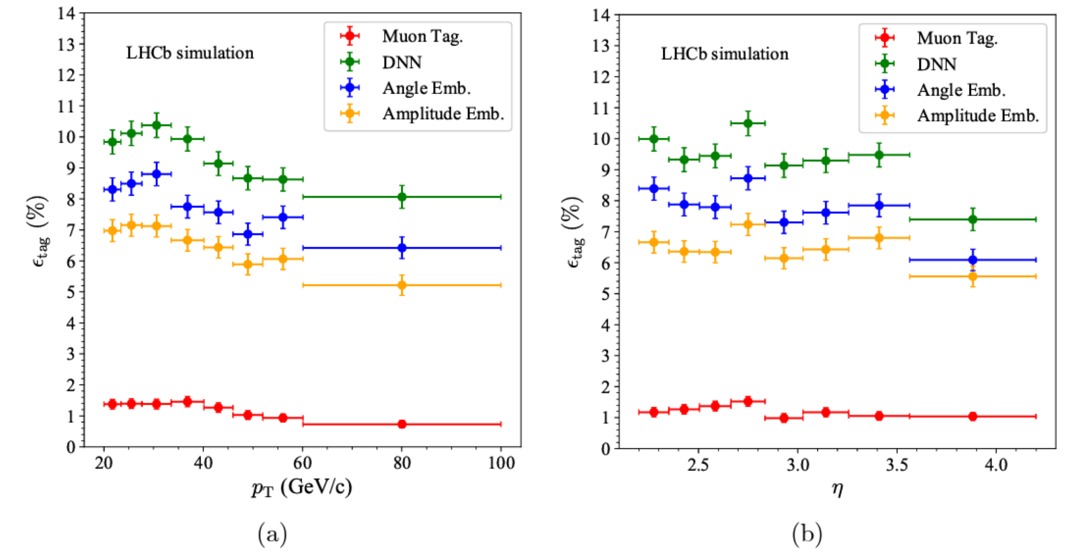


Figure 8: Tagging power ϵ_{tag} with respect to (a) jet p_{T} and (b) jet η for the *complete dataset*. The quantum algorithms perform slightly worse than the DNN, with the Angle Embedding circuit performing better than the Amplitude Embedding circuit.

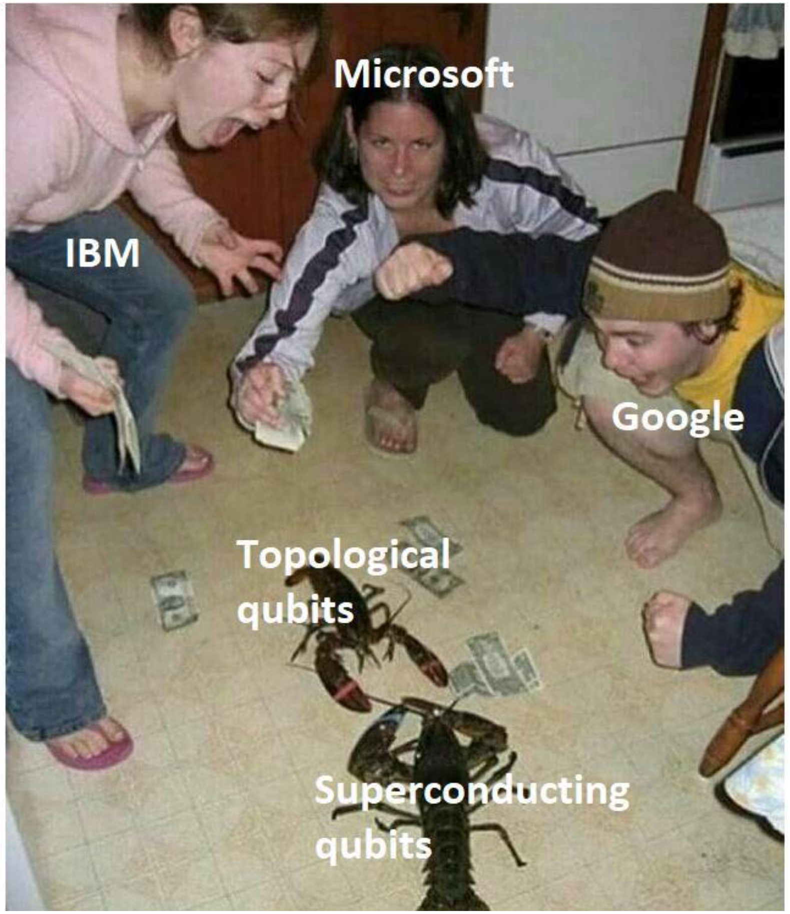
[arXiv:2202.13943]

Prospects and future applications

- High Energy Physics
- Complex Manufacturing and Industrial Design
- Logistics
- Finance and financial modelling
- Chemical and biological Engineering
- Pharmacy and drug development
- Artificial Intelligence
- Cybersecurity
- Material Science
- … and many more!

Future Predictions

- IBM currently have 433 qubits – expect to reach 1,000,000 qubits in **2027**
- Google has 53 qubits – expect to reach 1,000,000 qubits in **2029**
- Google announced their first error correcting, logical qubit in February 2023
- Many others are joining "the race"
- When do you think quantum computers will become "useful"?
- **Do you think it would happen at all?**



How can I participate?

CERN Quantum Technology Initiative

- First workshop on Quantum Computing in HEP at CERN in 2018
- Conference on Quantum Technologies in HEP last Novemeber
- Established a comprehensive R&D, academic and knowledge-sharing initiative for quantum technologies

- Currently focusing on:
  - Quantum computing and algorithms
  - Quantum theory and simulation
  - Quantum sensing, metrology and materials
  - Quantum communications and networks

QUANTUM TECHNOLOGY INITIATIVE

- Collaborating with universities and institutions all over the world with a lot of them being in Europe
- In 2021, CERN became a quantum hub in partnership with the IBM Q-Network
- Quantum Technology Initiative Journal Club – meeting on Thursdays
- https://quantum.cern/

Qiskit Quantum Developer Certificate

- Certification awarded for learning quantum computation using Qiskit
- Demonstrates you having fundamental knowledge of quantum computing concepts
- Demonstrates you being able to create and execute quantum computing programs on IBM Quantum computers and simulators
- Defining, executing, and visualizing results of quantum circuits with different gates, etc.
- Useful if you want to demonstrate your ability in quantum computing
- https://www.ibm.com/training/certification/C0010300

Takeaways

- Quantum computing is an emerging field that's still far away from being useful in solving real life problems at an exponential rate

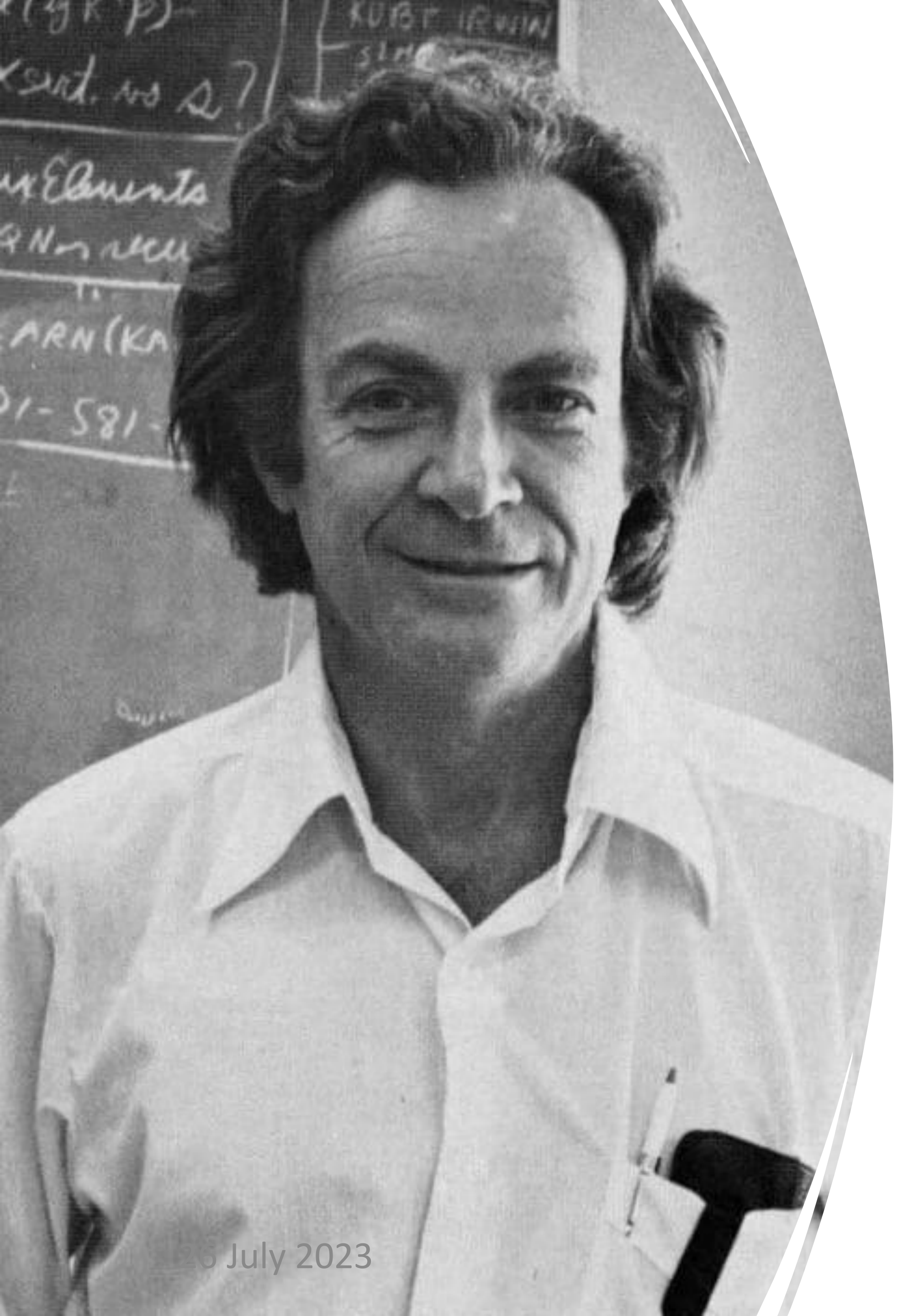
- Quantum computing is basically linear algebra and complex variables (unless you're working on developing the hardware)

- Current main problems revolve around error correction and increasing number of useful qubits

- When people first created the classical computer, they did not imagine how it would evolve. They probably never imagined the Internet and the horrors you can now find there

- Prospects are looking **okay** for the time being – just need to be cautiously optimistic and not blindly follow the hype train

- There are many ways you can get involved in quantum computing today!



“Nature isn't classical, dammit, and if you want to make a simulation of nature, you'd better make it quantum mechanical, and by golly it's a wonderful problem, because it doesn't look so easy.”

-Richard Feynman

Blooket

https://play.blooket.com

More resources

- Introduction to Quantum Computing and Quantum Hardware (Qiskit on YouTube)
- Qiskit textbook
- Xanadu textbook
- Quantum Machine Learning demonstrations (Pennylane)
- A course on Quantum Machine Learning (Github – Pennylane)
- 1 Minute Qiskit (Qiskit on YouTube)
- Why Did Quantum Entanglement Win the Nobel Prize in Physics? (PBS Spacetime on YouTube)
- What is Quantum Safe (IBM Technology on YouTube)

Thank you for your attention!

https://forms.gle/5NoXyLmBANAWA88d9

Bonus!

Quantum lingo cheat sheet

- **Noisy Intermediate-Scale Quantum (NISQ) era:** era with intermediate number of qubits (~100) that still have problems with noise/decoherence – era we are currently in

- **Fault tolerant computer:** Less affected by noise/decoherence

- **Fidelity:** a measure of how close the final quantum state of the real-life qubits is to the ideal case. The threshold for fault-tolerant quantum computing is over 99%

- **Quantum volume:** a metric that measures the capabilities and error rates of a quantum computer based on the size of the successfully running circuits. It was invented by IBM

- **Coherence time:** the length of time a quantum superposition state can survive

- **Transmon:** a superconducting loop-shaped qubit that can be created at extremely low temperatures

Cracking RSA encryption with a quantum computer

- Chinese scientists released a paper in Dec 2022 claiming to have cracked low-level RSA encryption using a hybrid of a quantum and classical computer

- 48-bit numbers using a 10-qubit quantum computer

- They combine classical lattice reduction factoring techniques (Schnorr's algorithm – not be confused with Shor's algorithm) with a quantum approximate optimization algorithm

- They calculated that it's possible to scale their algorithm for use with 2048-bit keys using a quantum computer with only **372 qubits**!

- IBM already has a 433-qubit quantum computer...

- Experts are saying these claims don't add up and shouldn't be scalable to higher qubit computer. Expect IBM to do some tests soon