Qiskit | Fall Fest

Quantum Computing Workshop

Wenyang Qian, Meijian Li, Juan Santos Suárez, Xoán Mayo López

Oct 21st, 2022, 11:30 - 18:30, Aula B, IGFAE,
University of Santiago de Compostela, Spain

Qiskit

IGFAE
Instituto Galego de Física de Altas Enerxías

USC
UNIVERSIDADE DE SANTIAGO DE COMPOSTELA

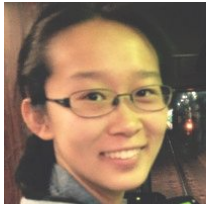
XUNTA DE GALICIA

Welcome to USC Qiskit Fall Fest!

Event schedule

Lectures	
Introduction to quantum computing basics	11:30 - 12: 20
Introduction to quantum algorithm	12:30 - 13: 20
Quantum Challenges	15:00 - 17:00
Discussion	17:00 - 17:45
Research talk in quantum computing	17:45 - 18:30


Wenyang Qian


Meijian Li


Juan Santos Suárez


Xoán Mayo López

Please feel free to ask questions to any of us

Additional info

In the Quantum Challenges, You will be doing problem solvings on 4 quantum computing problems using the Qiskit language. A personal computer is required to participate in the challenge. It is ok to ask help from mentors, but it is not allowed to collaborate on the solution.

Top 5 scores will be notified by us and given special prizes provided by the IBM Quantum team.

Optional survey email will be sent after the event. If possible, please provide your feedback, as we look to hold similar quantum computing event in the future.

Lecture: Introduction to quantum computing

Table of Content:

Development in Quantum computing

Quantum computing has come a long way in past 40 years

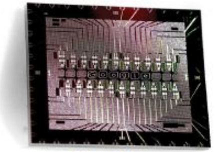
Feynman (1981)

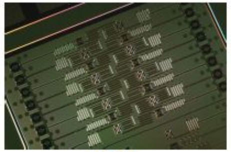
Toffoli Gate (1980) | Shor's Algo (1994) | Error Correction (1995) | Variational Eigensolver (2014) | Quantum Machine Learning (2017)

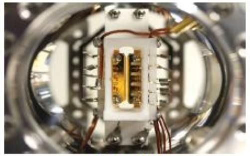
Currently, we are in the **Noisy Intermediate-Scale Quantum (NISQ)**: quantum bits (qubits) and quantum operations are substantially imperfect. Nonetheless,

Preskill 1801.00862 (2018)
Bharti et al. 2101.08448 (2021)

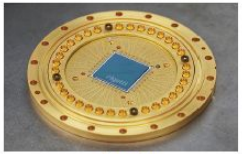
Image from Jong's talk (2022)


Google


IBM


Rigetti


Intel


IonQ

Qiskit

Cirq

XANADU

Qiskit

IGFAE Instituto Galego de Física de Altas Enerxías USC XUNTA DE GALICIA

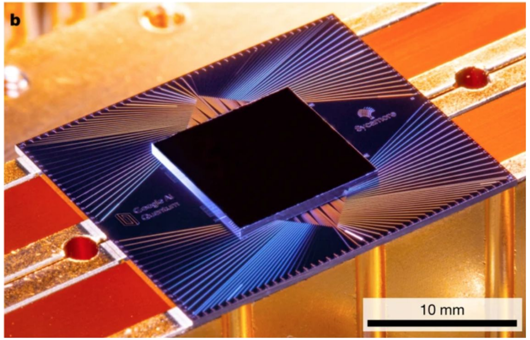
Towards Quantum advantage

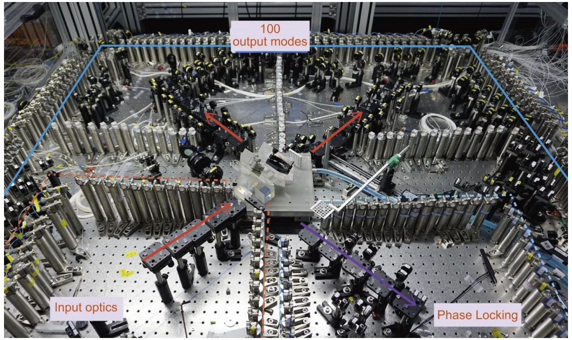
Quantum advantage: a purpose-specific computation that involves a quantum device and that can not be performed classically with a reasonable amount of time and energy resources.

Google AI Quantum 1910.11333 (2019)

- Sampling pseudo-random quantum circuit with 53-qubit Sycamore superconducting programmable chip
- Gaussian boson sampling with 76-photon Jiuzhang photonic quantum computer

UTSC, Jian-Wei Pan's group, 2012.01625 (2020)





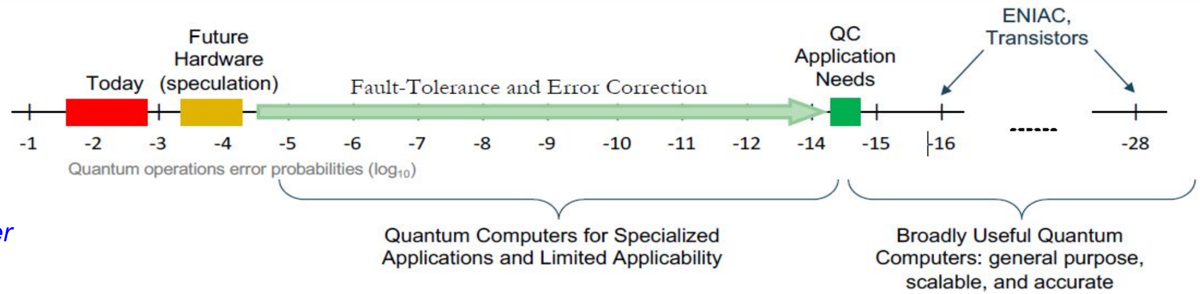
Recently, "Quantum Simulation for High Energy Physics", 2204.03381 (2022)

Why quantum computing?

“Nature isn't classical, dammit, and if you want to make a simulation of nature, you'd better make it quantum mechanical” (Richard Feynman)

- Many problems are inherently quantum mechanical
- Vast amount (exponential) of encoded information in a many-qubit state: Impossible to compete on classical computers; Nature does it automatically
- High scalability in quantum applications with efficient encoding
- Rapid progress in hardware, software, algorithms, benchmark

Image from Jong’s talk (2022)



Road to practical quantum computer

What is quantum computing/computer?

Classical computers

- classical bit (state): 0, 1
- classical gates: AND, OR, NOT, Bitwise logic gates
- deterministic nature

“switch”

Quantum computers

- quantum bit: **Qubit**, a two-level quantum system
- quantum gates: **Unitary operators**
- unique features from quantum mechanics:

Superposition

Entanglement

States collapse when measured

“ping pong ball”

Qubit (bra-ket notation)

Quantum bit, or qubit, can be in a superposition of state-0 and state-1, is a vector in a **two-dimensional complex vector space**. To represent quantum states of the qubit, we introduce **bra–ket notation**, or Dirac notation (1939). A **ket** denotes a column vector. For example, the two basis states in a single qubit are:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \; |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Together, we can represent the quantum state of the qubit completely, as **superposition** of the basis state,

$$|\Psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle\,,\, |\alpha|^2 + |\beta|^2 = 1$$

Likewise, a **bra** denotes a row vector (adjoint vector),

$$\langle 0| = \begin{pmatrix} 1 & 0 \end{pmatrix},\langle 1| = \begin{pmatrix} 0 & 1 \end{pmatrix} \qquad \langle \Psi| = \alpha^*\,\langle 0| + \beta^*\,\langle 1|$$

In this way, we can compute bra-ket as complex matrix multiplications. *Exercise: What are their evaluations*? $\langle 0|0\rangle\,,\langle 0|1\rangle\,,\langle 0|\psi\rangle\,,\langle \psi|\psi\rangle$

Qubit (bra-ket notation)

$$\langle 0|0\rangle = \begin{pmatrix}1 & 0\end{pmatrix}\begin{pmatrix}1\\0\end{pmatrix} = 1 \qquad \langle 0|1\rangle = \begin{pmatrix}1 & 0\end{pmatrix}\begin{pmatrix}0\\1\end{pmatrix} = 0$$

Basis orthonormality

$$\begin{aligned}\langle 0|\psi\rangle &= \langle 0|\left(\alpha\,|0\rangle + \beta\,|1\rangle\right)\\ &= \langle 0|\,\alpha\,|0\rangle + \langle 0|\,\beta\,|1\rangle\\ &= \alpha\,\langle 0|0\rangle + \beta\,\langle 0|1\rangle\\ &= \alpha\cdot 1 + \beta\cdot 0 = \alpha\end{aligned}$$

State projection

$$\begin{aligned}\langle\psi|\psi\rangle &= \left(\alpha^*\,\langle 0| + \beta^*\,\langle 1|\right)\left(\alpha\,|0\rangle + \beta\,|1\rangle\right)\\ &= \alpha^*\alpha\,\langle 0|0\rangle + \alpha^*\beta\,\langle 0|1\rangle + \beta^*\alpha\,\langle 1|0\rangle + \beta^*\beta\,\langle 1|1\rangle\\ &= |\alpha|^2\cdot 1 + 0 + 0 + |\beta|^2\cdot 1\\ &= 1\end{aligned}$$

State normalization

Note: these bracket form the **inner product** on the vector space, directly related to the quantum measurement. The **Hilbert space** is a complex vector space with an inner product.

Qubit (Bloch sphere visualization)

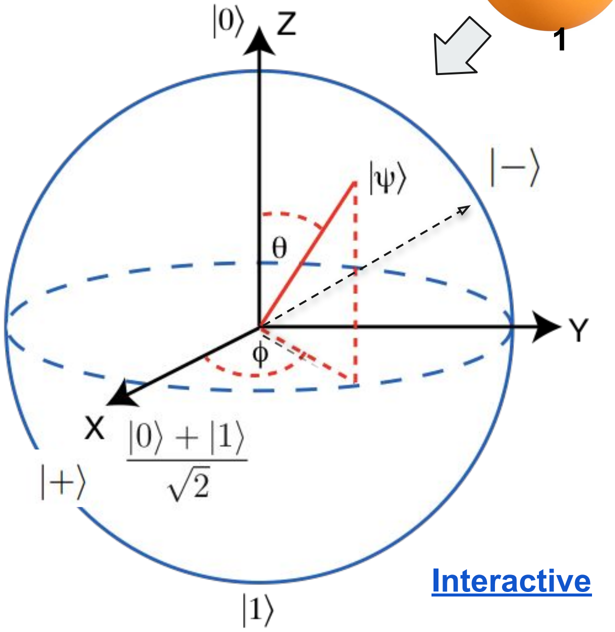
Equivalently, the qubit can also be represented as geometrically using the Bloch sphere (topologically 2-sphere), where *Φ* dictates the relative phase.

$$
\begin{aligned}
|\psi\rangle &= e^{i\gamma}\left(\cos\frac{\theta}{2}\,|0\rangle + e^{i\phi}\sin\frac{\theta}{2}\,|1\rangle\right) \\
&\equiv \cos\frac{\theta}{2}\,|0\rangle + e^{i\phi}\sin\frac{\theta}{2}\,|1\rangle
\end{aligned}
$$

Other popular set of measurement bases are:

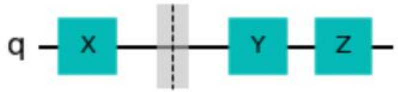
$$
|+\rangle \triangleq \frac{|0\rangle + |1\rangle}{\sqrt{2}},\; |-\rangle \triangleq \frac{|0\rangle - |1\rangle}{\sqrt{2}}
$$

$$
|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle = \frac{\alpha+\beta}{\sqrt{2}}\,|+\rangle + \frac{\alpha+\beta}{\sqrt{2}}\,|-\rangle
$$



Interactive

Quantum gates are unitary operators



Quantum gates are what we used to modify the state of the qubit. Unitary operators are operators on Hilbert space that preserves the inner product. They are written as unitary matrices, satisfying $U^\dagger U = UU^\dagger = I$

<u>All quantum gates are unitary and any unitary operator is a valid quantum gate!</u>

Unlike classical gates, there are many non-trivial single-qubit gates, i.e., X, Z, Hadamard (H), Phase (P) gates. We apply them on quantum state using matrix multiplication. *(short exercise)*

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, X\begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \beta \\ \alpha \end{pmatrix}, X|0\rangle = |1\rangle, X|1\rangle = |0\rangle$$

$$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, Z|0\rangle = |0\rangle, Z|1\rangle = -|1\rangle$$

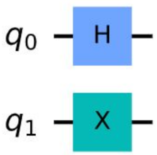
$$H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, H|0\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}}, H|1\rangle = \frac{|0\rangle - |1\rangle}{\sqrt{2}}$$

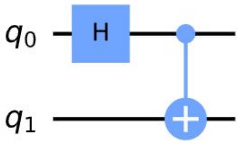
$$H^2 = I$$

$$P(\lambda) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\lambda} \end{pmatrix}$$

Note: Quantum gates are always reversible, unlike classical

Multi-qubit states





The quantum advantage lies in multiple qubit states.

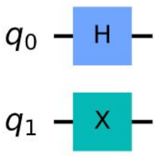
The quantum controlled-NOT gate (CNOT, or CX) is the most useful two-qubit gate. The gate acts on two qubits, A (control) and B (target). If the control qubit is in state 1 (True), the CNOT gate negate the value in the target qubit. It enables the simplest form of **quantum entanglement**.

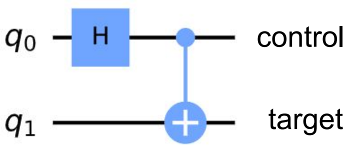
$$CNOT\,|A, B\rangle = |A, B \oplus A\rangle$$
$$|00\rangle \rightarrow |00\rangle, |01\rangle \rightarrow |01\rangle, |10\rangle \rightarrow |11\rangle, |11\rangle \rightarrow |10\rangle$$

$$CNOT = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Exercise: what is quantum state of the two circuits above?

Multi-qubit states





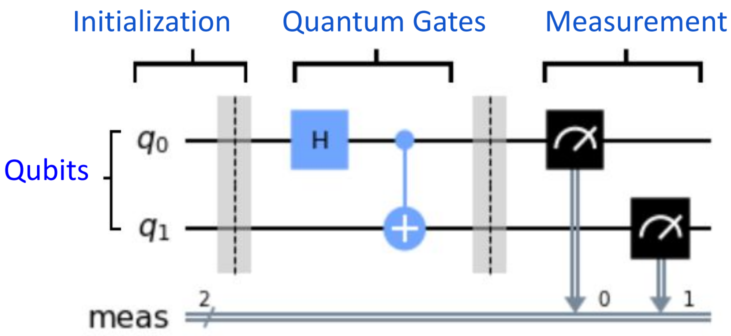
$$H_0 X_1 |00\rangle = H|0\rangle \otimes X|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |1\rangle = \frac{1}{\sqrt{2}}|01\rangle + \frac{1}{\sqrt{2}}|11\rangle$$

$$CX_{0,1} H_0 |00\rangle = CX_{0,1} \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |0\rangle = CX_{0,1} \frac{1}{\sqrt{2}}(|00\rangle + |10\rangle) = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$

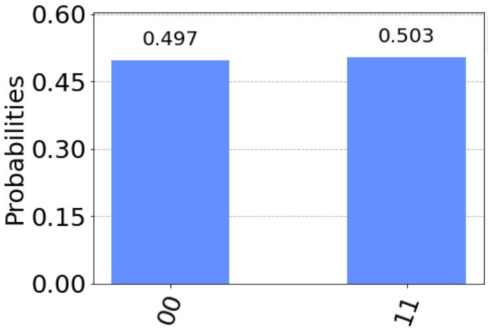
Note: the quantum state for the second circuit cannot be decomposed into product of two quantum states anymore. It is an entangled state. In particular, this one is a Bell state.

One can also calculate the quantum state using matrix form of the operators (intro).

Quantum circuit and measurement



Quantum circuit



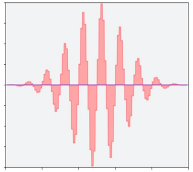
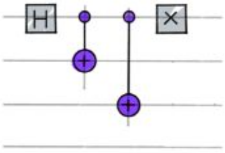
Measurement outcome (1024 shots)

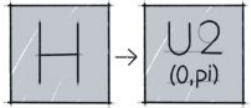
What we see so far are quantum circuits!

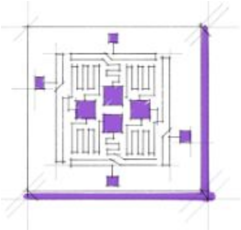
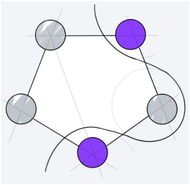
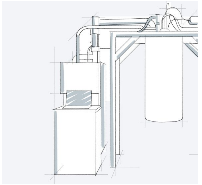
- Each horizontal line represents the evolution of a qubit (from left to right)
- Each measurement **collapses** the wave function and we obtain either 0 or 1 in the computation basis
- In practice, measurements are statistical outcomes by running the same circuit repeatedly for thousands of times (or shots)

Qiskit: open-source quantum computing framework

“Qiskit [kiss-kit] is an open-source SDK for working with quantum computers at the level of **pulses, circuits, and application modules**.”



Applications in natural science, optimization, quantum machine learning, finance.

Particularly, Qiskit provides unique access to various quantum backends, including cloud quantum computers and noise simulations.

qiskit.opflow library (Quantum algebra)

Opflow library allows easy manipulation of quantum algebra and connects to quantum experiments.

- **Qubit states:** $|0\rangle \rightarrow \texttt{Zero}, \quad |1\rangle \rightarrow \texttt{One}, \quad |+\rangle \rightarrow \texttt{Plus}, \quad |-\rangle \rightarrow \texttt{Minus}$
- **Quantum operators:** $\text{X-gate} \rightarrow \texttt{X}, \quad \text{Y-gate} \rightarrow \texttt{Y}, \quad \text{Z-gate} \rightarrow \texttt{Z}, \quad \text{H-gate} \rightarrow \texttt{H}, \quad \text{I-gate} \rightarrow \texttt{I}$
- **Operations:**

$X|0\rangle \rightarrow \texttt{X @ Zero}, \quad H|+\rangle \rightarrow \texttt{H @ Plus}$ — Single-qubit state

$|0\rangle \otimes |0\rangle \equiv |00\rangle \rightarrow \texttt{Zero\^{}Zero}, \quad |+-\rangle \rightarrow \texttt{Plus\^{}Minus}$ — Multi-qubit state

$|0\rangle + \frac{1}{2}|1\rangle \rightarrow \texttt{Zero + 0.5 * One}, \quad |+\rangle - i|-\rangle \rightarrow \texttt{Plus - 1j * Minus}$ — Superposition

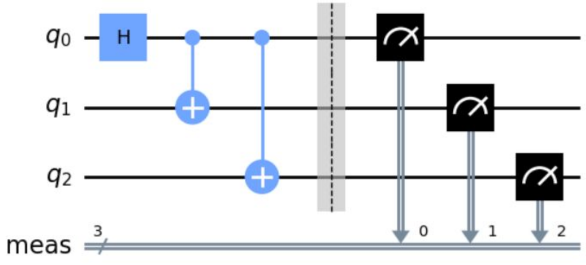
$XY \rightarrow \texttt{X @ Y}, \quad HX|+\rangle \rightarrow \texttt{H @ X @ Plus}$ — Gate composition

$\langle +|H|+\rangle \rightarrow \texttt{(}\sim \texttt{Plus @ H @ Plus).eval()}$ — Expectation value

qiskit.circuit library (Building quantum circuit)

The fundamental element of quantum computing is the quantum circuit. In Qiskit, this core element is represented by the circuit library. The *QuantumCircuit* class is particularly useful.

```
from qiskit import QuantumCircuit                # calling library

circuit = QuantumCircuit(3)                      # create a circuit with 3 qubits
                                                 # (q0, q1, q2)

circuit.h(qubit=0)                               # apply a Hadamard gate to q0
                                                 # qubits are always indexed from 0

circuit.cx(control_qubit=0, target_qubit=1)      # apply a controlled-not gate with
                                                 # control on q0 and target on q1

circuit.cx(0, 2)                                 # apply a controlled-not gate with
                                                 # control on q0 and target on q2

circuit.measure_all()                            # apply measurement and map results
                                                 # to classical bits

circuit.draw('mpl')                              # draw circuit with output style "mpl"
```



qiskit.Aer library (IBM Quantum simulator & device)

The qiskit.Aer library enables quantum simulation with versatile simulators.

Exact simulation
(testing purpose)

```
from qiskit import Aer # import Aer library for simulation
```

```
sv_backend = Aer.get_backend('statevector_simulator')
qasm_backend = Aer.get_backend('qasm_simulator')
```

Statistical simulation
(ideal quant computer)

```
job = execute(circuit, backend=qasm_backend, shots=1024)  # execute circuit on specified backend with measurement shots
result = job.result()                                     # acquire result from job
counts = result.get_counts()                              # acquire counts
print(counts)
```

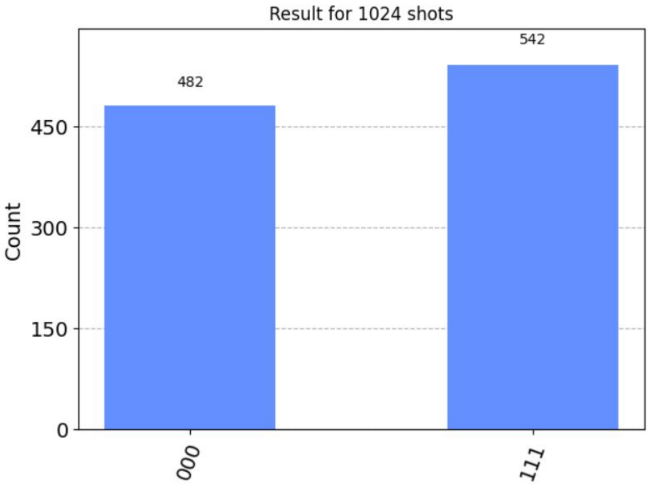
```
{'111': 542, '000': 482}
```

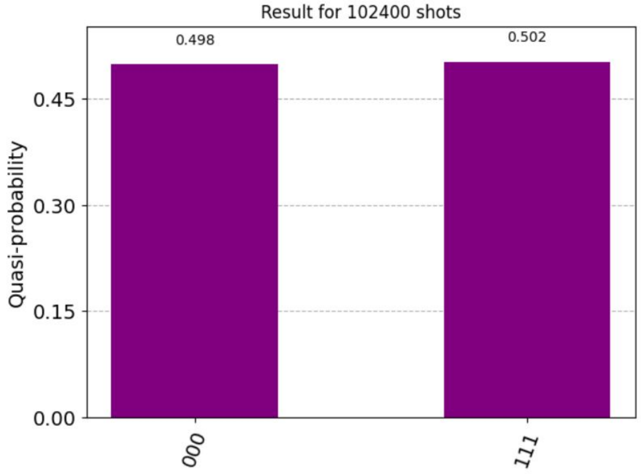
This Python dictionary indicates, out of 1024 measurements, we obtain the final state |111> for 512 times and |000> for 482 times. As the number of measurement increases, we expect to get a more represented results for the true quantum state.

qiskit.visualization library (Visualization toolkit)

The visualization module contain functions that visualizes measurement outcome counts, quantum states, circuits, pulses, devices and more.

```
from qiskit.visualization import plot_histogram
plot_histogram(counts, title='Result for 1024 shots')
```

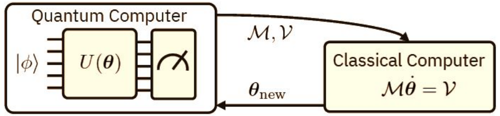




Quantum computing in near future

Several main directions

- **Variational Approaches**: Variational Quantum Eigensolver (VQE), hybrid optimization algorithm, many variants, widely-used in quantum chemistry, lead to Quantum Machine Learning



Peruzzo et al. 1304.3061 (2013)
Bharti et al. 2101.08448 (2021)

- **Decomposition Approaches**: Quantum Simulation Algorithms, prepare, evolve, fourier transform, measure to find quantum state of the system **(see Research talk at 17:45 by Meijian)**

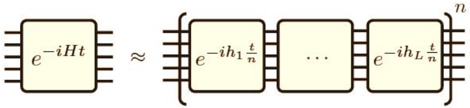


Image from Miessen's talk at QGSS 2022

- **Quantum computing algorithms (see Lecture B at 12:30 by Juan)**

IBM Quantum public backend

ibm_nairobi

Details

7 Qubits

32 QV

2.6K CLOPS

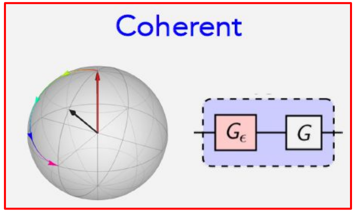
Status:	Online	Avg. CNOT Error:	1.024e-2
Total pending jobs:	27 jobs	Avg. Readout Error:	2.923e-2
Processor type ⓘ:	Falcon r5.11H	Avg. T1:	104.68 us
Version:	1.0.24	Avg. T2:	62.85 us
Basis gates:	CX, ID, RZ, SX, X	Providers with access:	1 Providers ↓
Your usage:	2952 jobs		

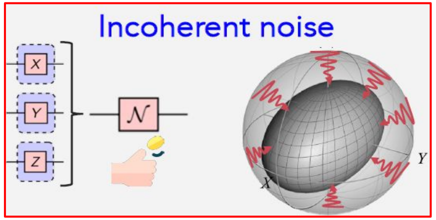
Overall performance:

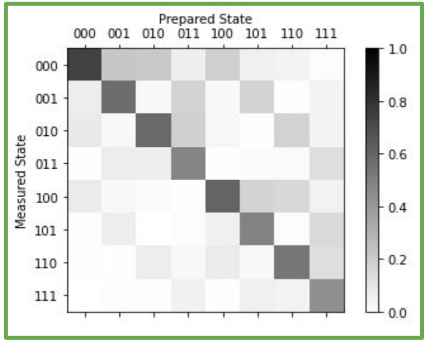
7 qubits, comparable to a circuit size of 5 by 5, 2.6K CLOPS

T1 relaxation time (|1> => |0>)

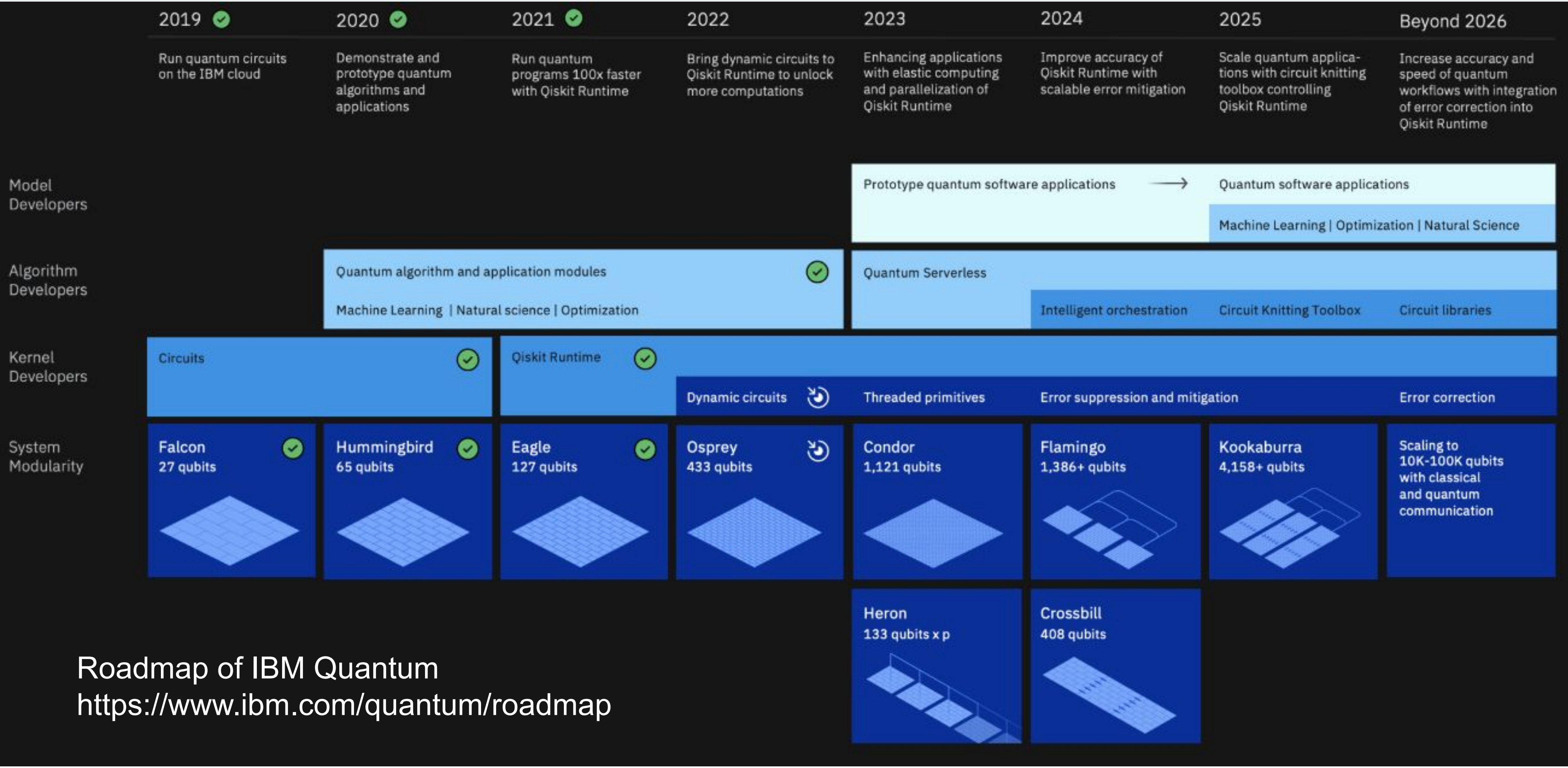
T2 dephasing time (|+> => |->)







Two images from Minev's talk at QGSS 2022



Roadmap of IBM Quantum
https://www.ibm.com/quantum/roadmap

Thank you!

Qiskit

IGFAE
Instituto Galego de Física de Altas Enerxías
USC
XUNTA DE GALICIA

Where to find more quantum computing resources?

We only scratched the surface of quantum computing basics. If you are interested in learning more, I highly recommend the following resources:

Qiskit Online Textbook https://qiskit.org/textbook/preface.html

Quantum Computation and Quantum Information (Isaac Chuang and Michael Nielsen)

Qiskit
qiskit.slack.com

Qiskit
Github

Qiskit/**qiskit.org**

The Qiskit official website

53 Contributors · 191 Issues · 37 Discussions · 81 Stars · 92 Forks

IBM Quantum Challenge Fall 2022
(Nov 11st 3PM - Nov 18th 3PM)