Quantum Neural Networks and Applications to Agriculture

Pabitra Mitra

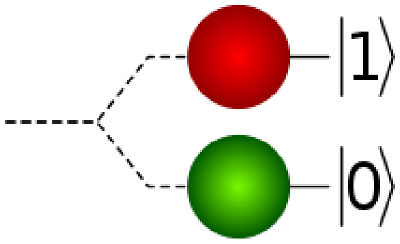
CSE, IIT Kharagpur

pabitra@cse.iitkgp.ac.in

Outline

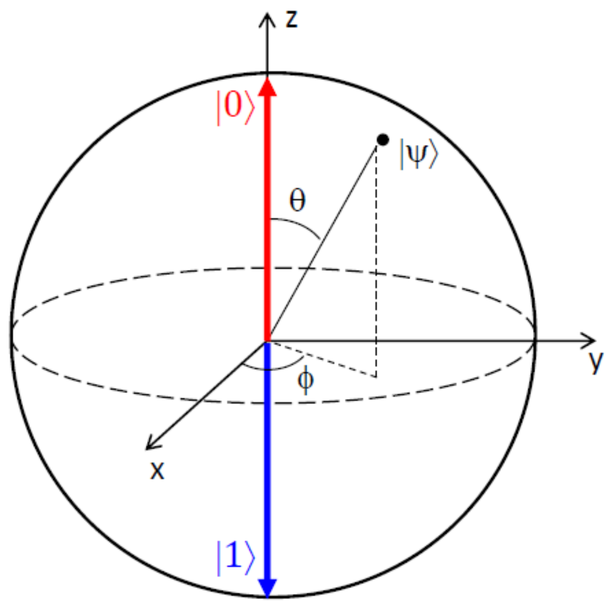
- Qubits
- Entanglement
- Quantum gates
- Quantum circuits
- Quantum neural networks
- Hybrid quantum neural networks
- Our research

Qubits



- Two state quantum mechanical system
  - basic unit of quantum information
  - quantum version of the classic binary bit
  - physically realized with a two-state device
- Examples:
  - spin of electrons – up/down
  - polarization of photons – left/right
- Coherent superposition of multiple states simultaneously
  - quantum mechanical property

Mathematical definition



Bloch sphere

- **Qu**antum **bi**nary digi**t** (qubit)

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

- State $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ with $|\alpha|^2 + |\beta|^2 = 1$ can be represented by

$$|\psi\rangle = \begin{bmatrix} \cos\frac{\theta}{2} \\ e^{i\phi}\sin\frac{\theta}{2} \end{bmatrix}$$

for θ in $[0, \pi]$ and ϕ in $[0, 2\pi]$

- Standard basis is z-basis (North/South axis)

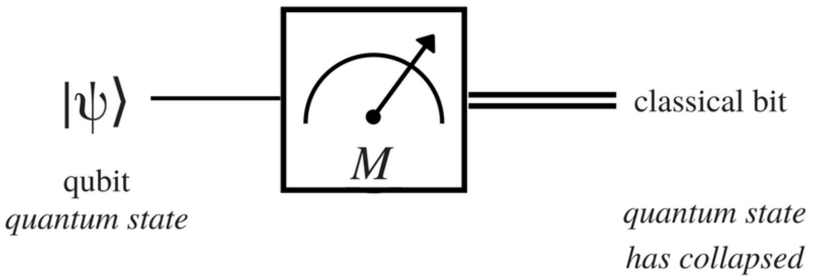
$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

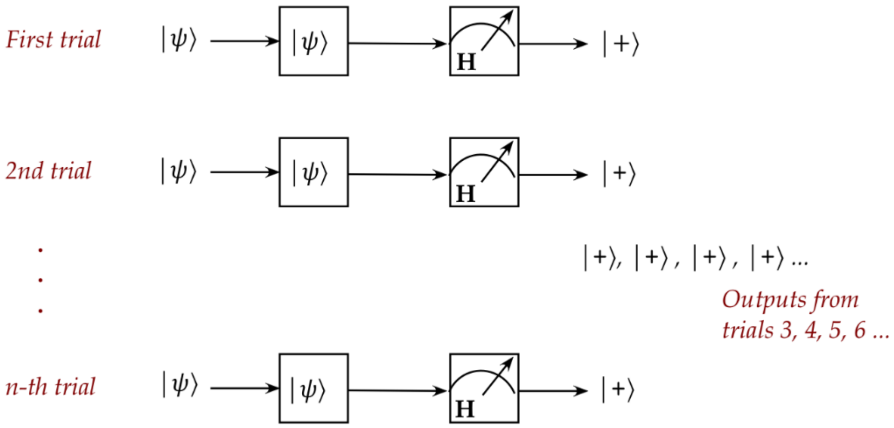
Coherent superposition – pure state

Vectors in a Hilbert space over complex field

Born rule – probability amplitudes

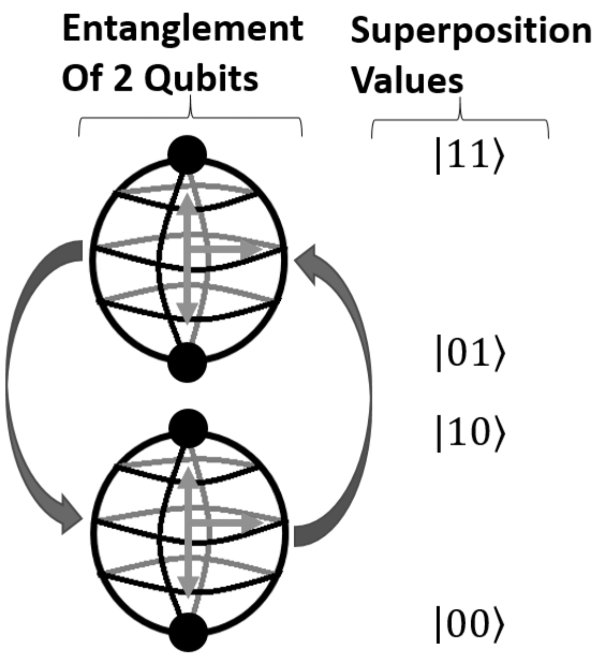
Measurements on Qubit

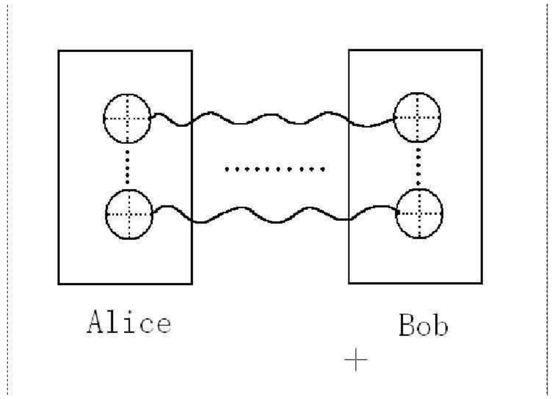




Expectations as probability amplitudes

Multiple Qubits: Entanglement





Bell state

$$|\Psi\rangle = \frac{1}{\sqrt{2}}\left(|00\rangle + |11\rangle\right)$$

Multiple Qubits: Entanglement

An n-qubit system can exist in any superposition of the 2^n basis states

$$c_0|00\ldots00\rangle + c_1|00\ldots01\rangle + \cdots + c_{2^n-1}|11\ldots11\rangle\,, \qquad \sum_{i=0}^{2^n-1} |c_i|^2 = 1$$

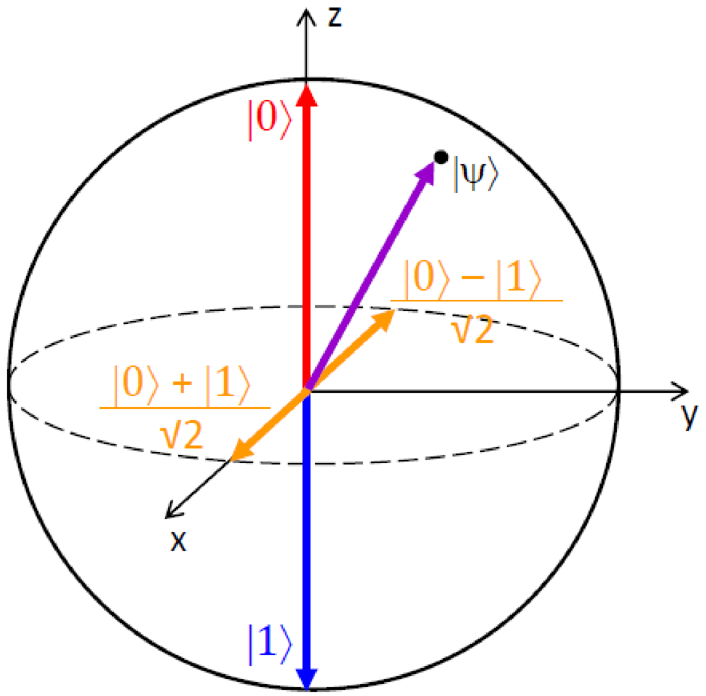
If such a state can be represented as a tensor product of individual qubit states then the qubit states are **not entangled**. For example:

$$\underbrace{\left(\frac{1}{\sqrt{8}}|00\rangle + \frac{\sqrt{3}}{\sqrt{8}}|01\rangle + \frac{1}{\sqrt{8}}|10\rangle + \frac{\sqrt{3}}{\sqrt{8}}|11\rangle\right)}_{2^n \text{ probability amplitudes}} = \underbrace{\left(\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle\right)\otimes\left(\frac{1}{2}|0\rangle + \frac{\sqrt{3}}{2}|1\rangle\right)}_{2n \text{ probability amplitudes}}$$

$$\left(\frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle\right) \neq (a|0\rangle + b|1\rangle)\otimes(c|0\rangle + d|1\rangle)$$

Quantum Gates: Unitary Transformations

1-qubit logic gates: rotations around x, y and z axes



Pauli X - rotation by π around x axis
- implements NOT gate

Hadamard - rotation by $\pi/2$ around y axis and then by π around x axis
- creates equal superposition when applied to $|0\rangle$ or $|1\rangle$

We can define infinitely many logic gates operating on a single qubit

Any unitary 2x2 matrix (rotation) is a logic gate

NOT and Hadamard Gates

$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$|0\rangle \;—\; \boxed{X} \;—\; |1\rangle$$

$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$|1\rangle \;—\; \boxed{X} \;—\; |0\rangle$$

Hadamard gate: creates superposition

$$\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} 1 \\ 0 \end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ 0 \end{bmatrix} + \frac{1}{\sqrt{2}}\begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$|0\rangle \;—\; \boxed{H} \;—\; \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$$

$$\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} 0 \\ 1 \end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ 0 \end{bmatrix} - \frac{1}{\sqrt{2}}\begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$|1\rangle \;—\; \boxed{H} \;—\; \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle$$

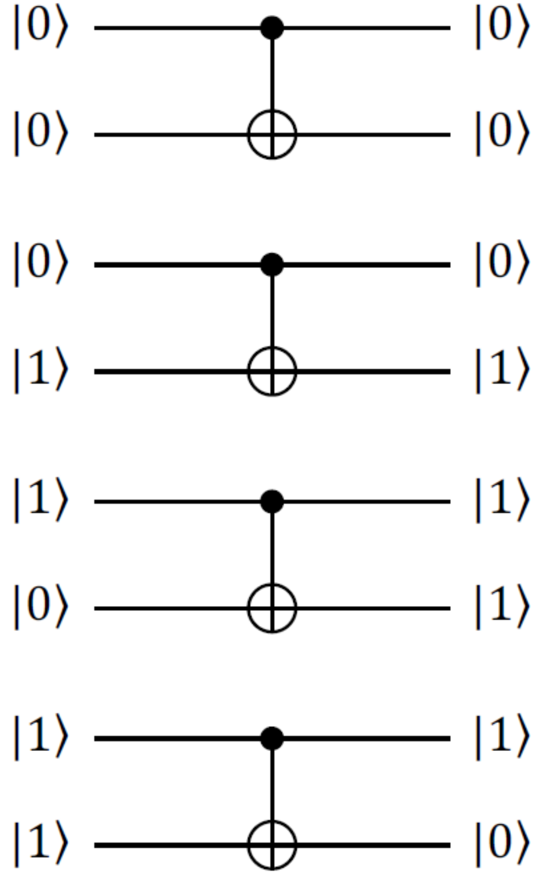
Controlled NOT (CNOT) – 2qubit gate

Truth table for the CNOT gate

A	B	X	Y
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0

Quantum CNOT gate

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$



Quantum Logic Gates

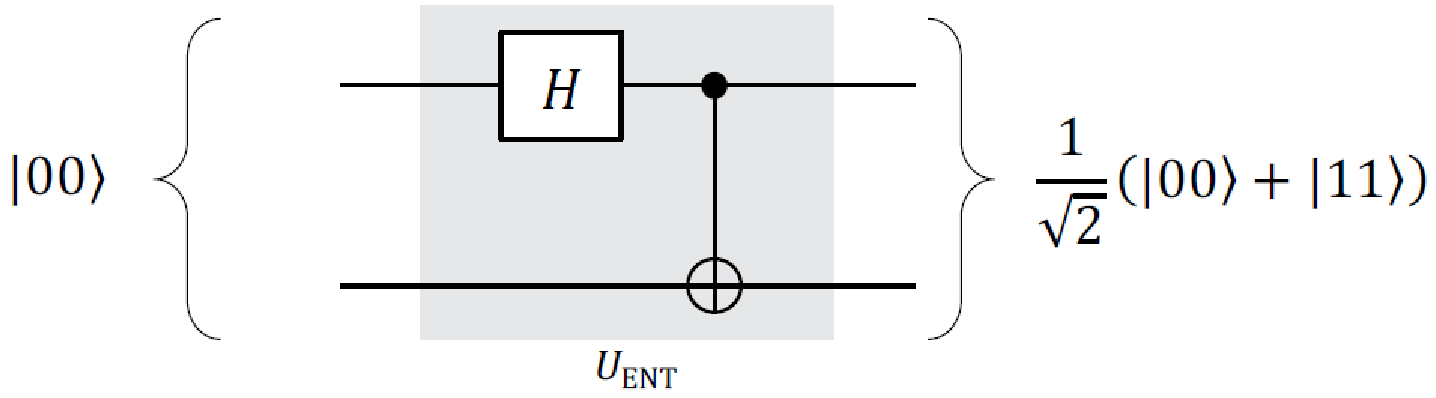
Rotation RY (Parameterised) Gates

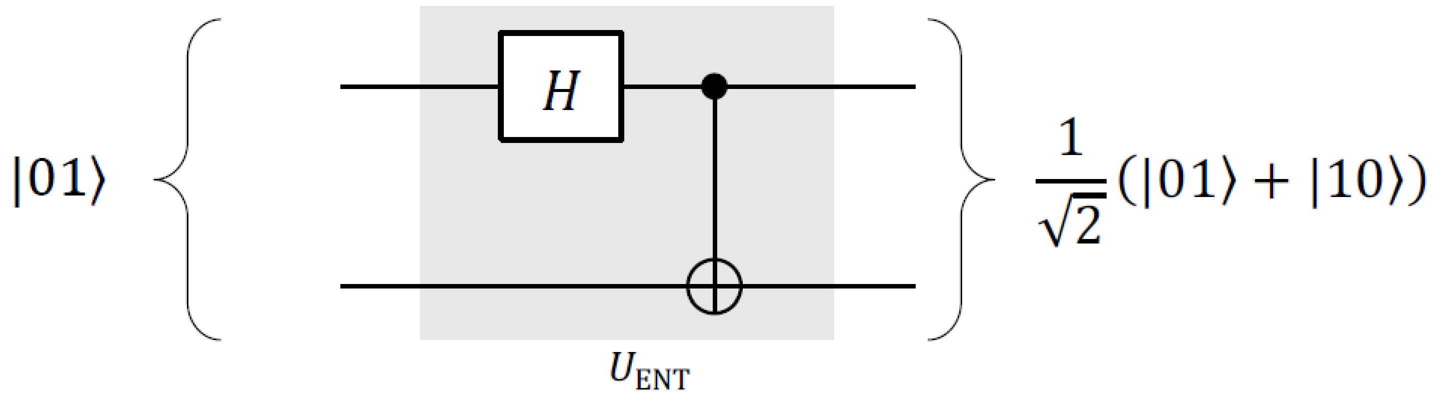
$$R_y(\theta) = \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & -\sin\left(\frac{\theta}{2}\right) \\ \sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{pmatrix}$$

CNOT and SWAP are universal 2-qubit gates

Operator	Gate(s)	Matrix
Pauli-X (X)	$\boxed{X}$	$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
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)		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$
Controlled Z (CZ)	$\boxed{Z}$	$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$
SWAP		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
Toffoli (CCNOT, CCX, TOFF)		$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$

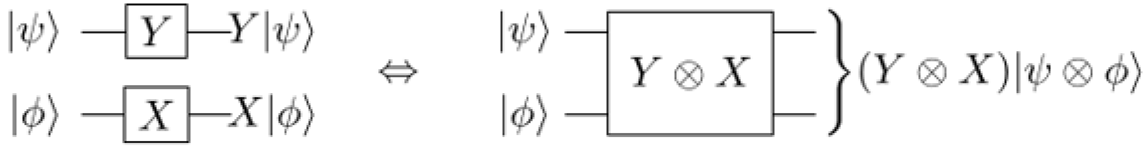
Creation of Entangled States with Gates





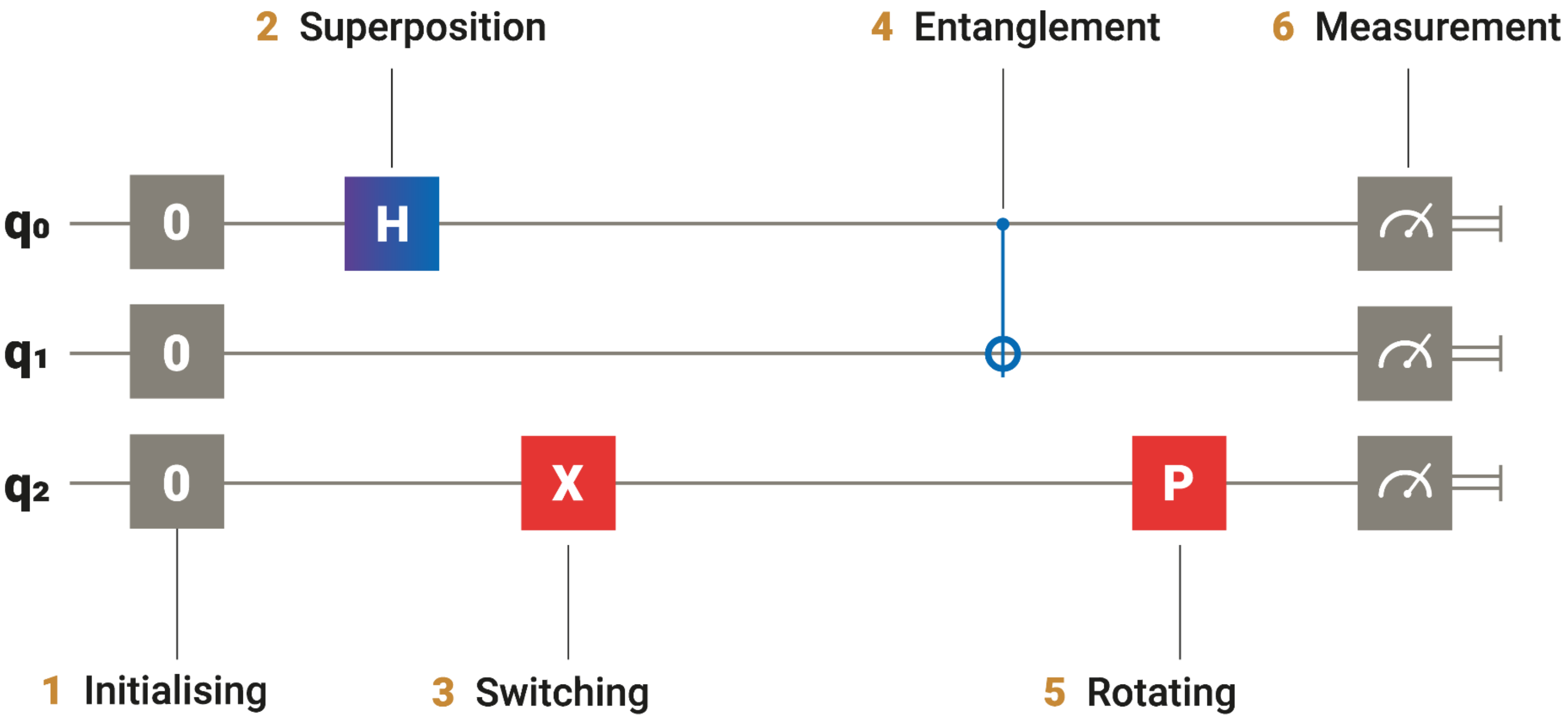
Wiring the Gates

Serial wiring



Parallel wiring

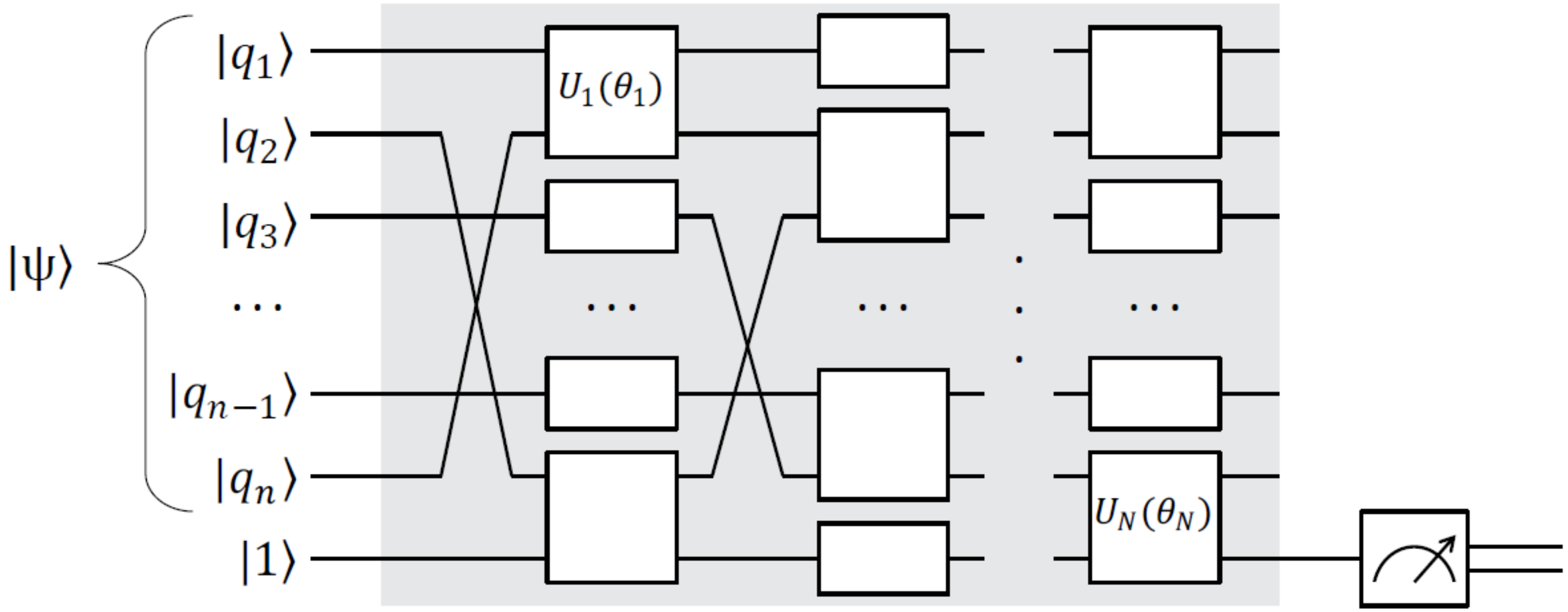
Quantum Circuits



Quantum Circuit Rules

- Not all quantum circuits are valid
- Gates are applied sequentially left to right – state evolution
- Wires represent identity matrices (“do nothing”)
- There are no loops
- Number of qubits is preserved

Parameterised Quantum Circuits

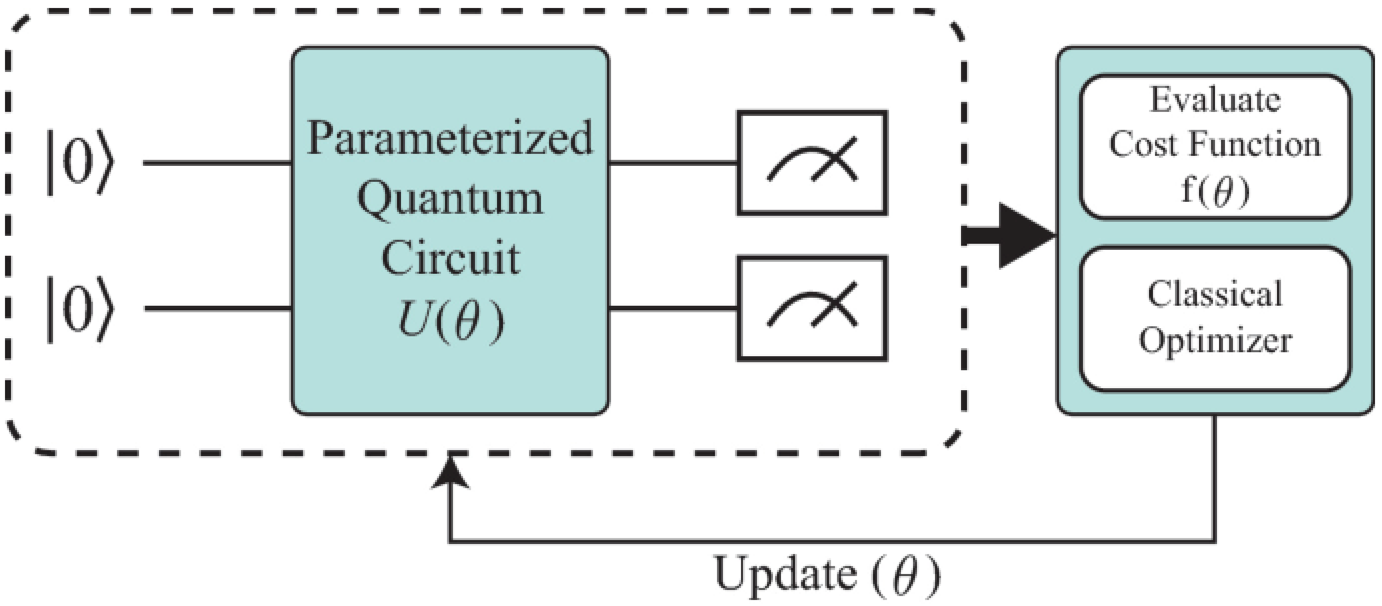


$|\psi\rangle = |q_1, q_2, \ldots, q_n\rangle$ Input qubit vector

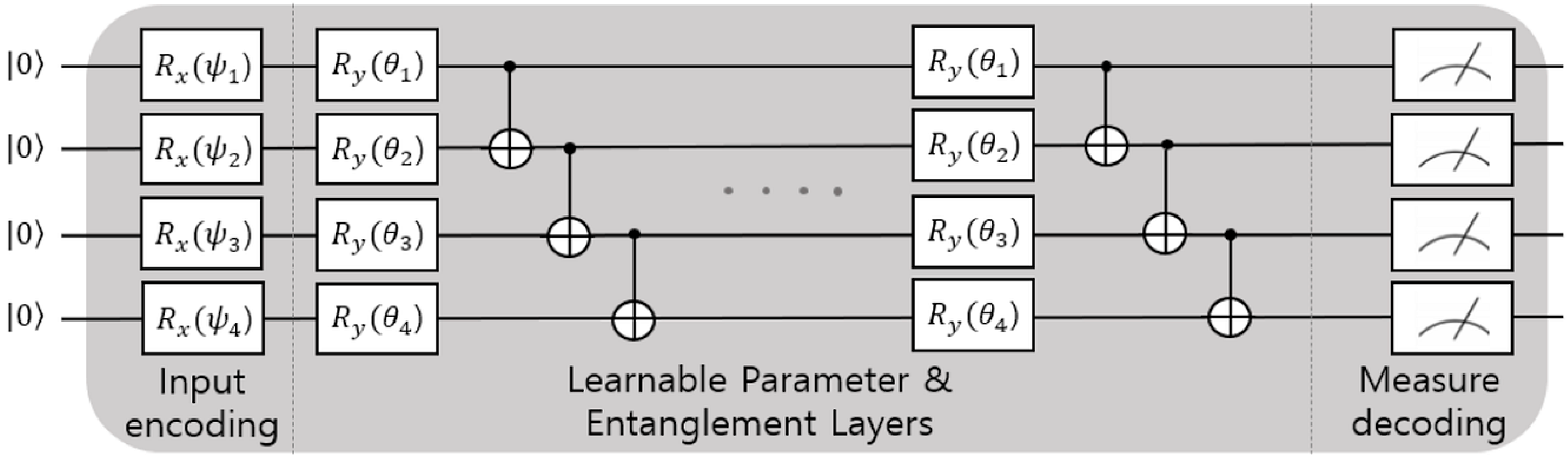
$U(\theta)$ 1 and 2 bit Unitaries

Parameters θ are adjusted during learning such that the measurement on the readout qubit tends to produce the desired label for $|\psi\rangle$.

Variational Quantum Circuits



Quantum Neural Networks



How to train your QNN?

Two approaches:

I. *Simulator-based*

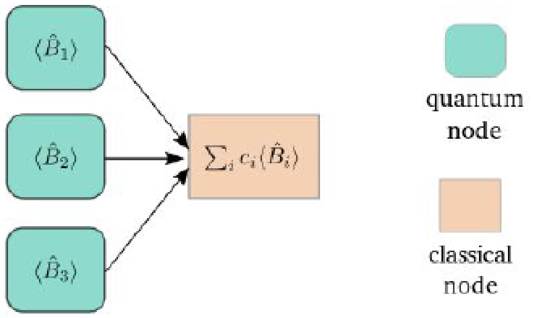
- Build simulation **inside existing classical library**
- Can leverage existing optimization & ML tools
- Great for small circuits, but **not scalable**

STRAWBERRY FIELDS

QuantumFlow

II. *Hardware-based*

- **No access to quantum information**; only have measurements & expectation values
- Needs to work as hardware becomes more powerful and **cannot be simulated**



Gradient of Quantum Circuits

- Training strategy: use gradient descent algorithms.
- Need to compute gradients of variational circuit outputs w.r.t. their free parameters.
- How can we compute gradients of quantum circuits when even simulating their output is classically intractable?

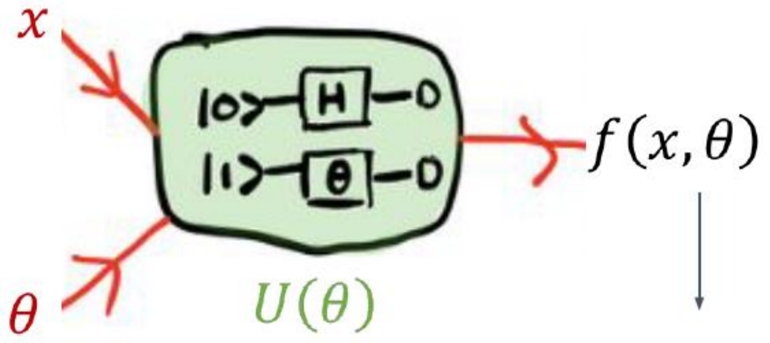
Computing the Gradient

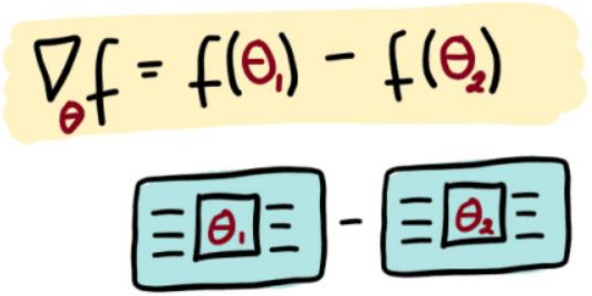
- No cloning of information between the layers
  - Unlike classical neural networks
- Fan-out unitary gate
- Ancilla bit

Parameter Shift Trick

Use the same device to compute a function and its gradient

- "Parameter shift" differentiation rule: gives **exact gradients**





$$\partial_\theta f(\theta) = c[f(\theta + s) - f(\theta - s)]$$

- Minimal overhead to compute gradients vs. original circuit
- Optimize circuits using **gradient descent**
- Compatible with classical backpropagation: hybrid models are **end-to-end differentiable**

Training Methods

- Gradient-based Optimisation
- Bayesian Optimisation
- Particle Swarm Optimisation
- Genetic Algorithm

Jin-Guo Liu and Lei Wang (2018)
Differentiable Learning of Quantum Circuit Born Machine
https://arxiv.org/abs/1804.04168

X. Gao, Z.-Y. Zhang and L.-M. Duan (2017)
An efficient quantum algorithm for generative machine learning
https://arxiv.org/abs/1711.02038

D. Zhu, N. M. Linke *et al* (2018)
Training of quantum circuits on a hybrid quantum computer
https://arxiv.org/abs/1812.08862

Marcello Benedetti, Delfina Garcia-Pintos *et al* (2019)
A generative modelling approach for benchmarking and training shallow quantum circuits
https://arxiv.org/abs/1801.07686

Alexei Kondratyev and George Giorgidze (2017)
Evolutionary Algos for Optimising MVA, Risk, December 2017
https://www.risk.net/cutting-edge/banking/5374321/evolutionary-algos-for-optimising-mva

Davide Venturelli and Alexei Kondratyev (2019)
Beyond Markowitz with Quantum Annealing, Risk, June 2019
https://www.risk.net/asset-management/6685986/beyond-markowitz-with-quantum-annealing

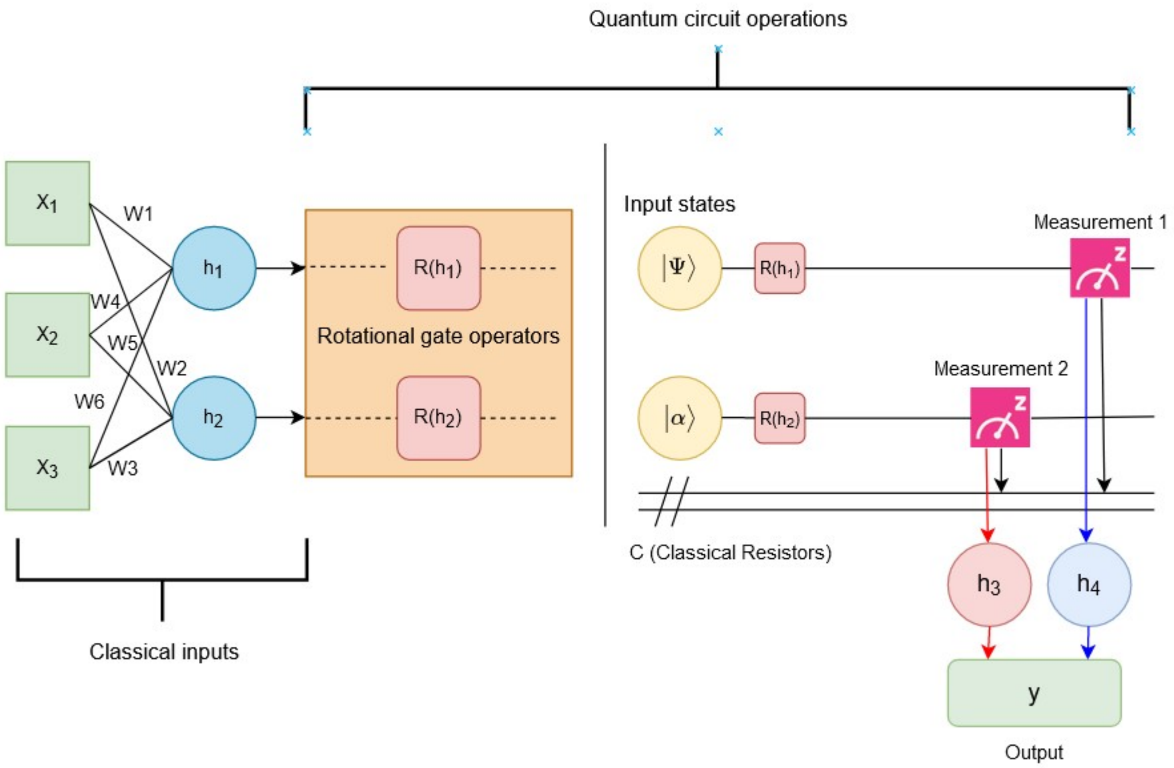
Alexei Kondratyev (2020)
Non-Differentiable Learning of Quantum Circuit Born Machine with Genetic Algorithm
https://ssrn.com/abstract=3569226

QNN Summary

- Decision choices
  - Input encoding – quantum embedding
  - Parameterised circuit topology
  - Cost function

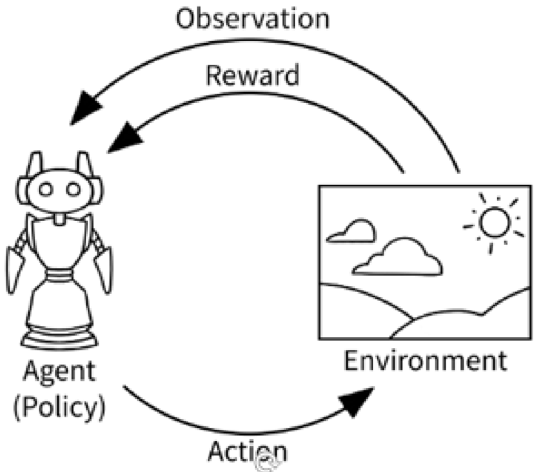
- Advantages
  - More expressive power with less number of nodes
  - Parallel computation

Hybrid Quantum Neural Networks



M. Das, A. Naskar, P. Mitra, B. Basu (2024) **Shallow quantum neural networks (SQNNs) with application to crack identification,** Applied Intelligence

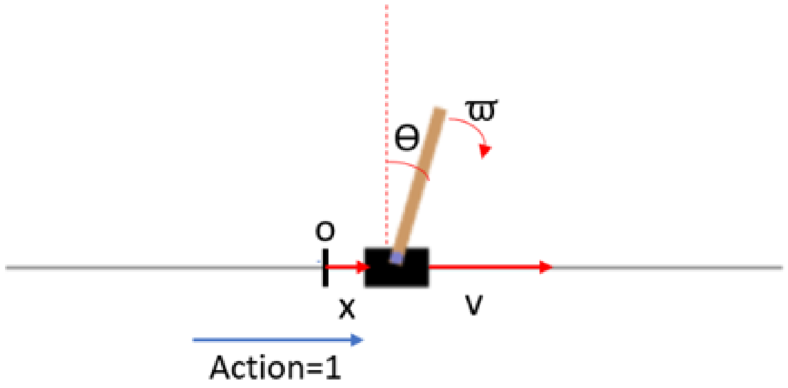
QNN for Optimal Control



Taken from openAI gym

Investigation of the performance and behavioural differences that Hybrid Quantum-Classical Reinforcement Learning agents have in comparison to Classical RL agents, specifically with respect to the sample efficiency and result stability of the relevant models

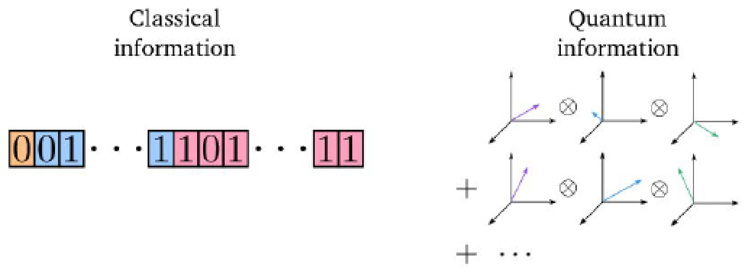
CartPole Stabilization problem



E. Mitchell, B. Basu, P. Mitra (2024) Performance analysis of a quantum-classical hybrid reinforcement learning, ICLR 2024, Vienna.

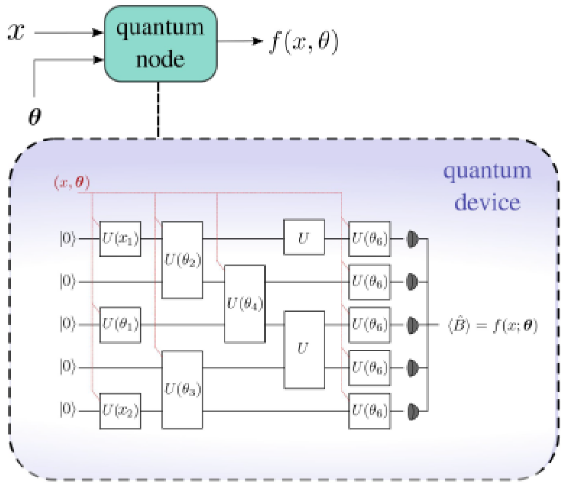
Quantum Node – A Programming Construct

Classical and quantum information are distinct

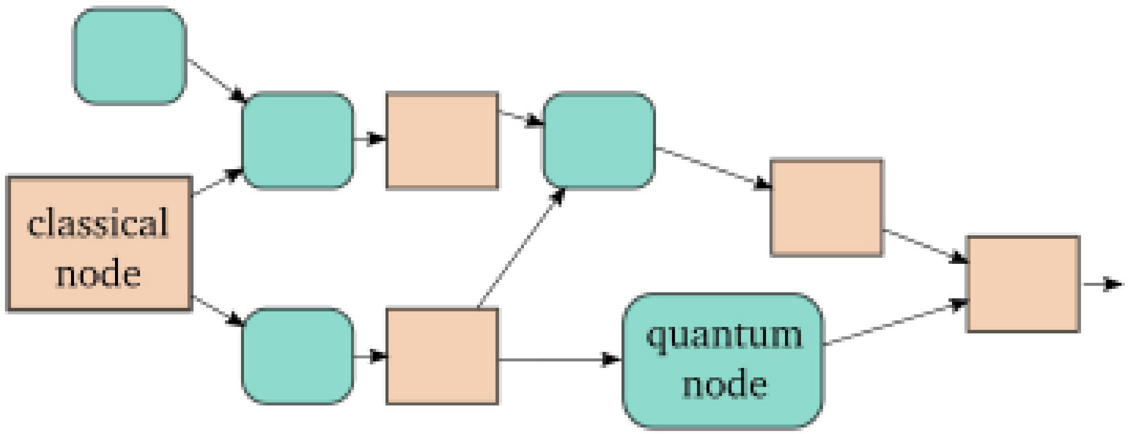


QNode: common interface for quantum and classical devices

- Classical device sees a callable parameterized function
- Quantum device sees fine-grained circuit details



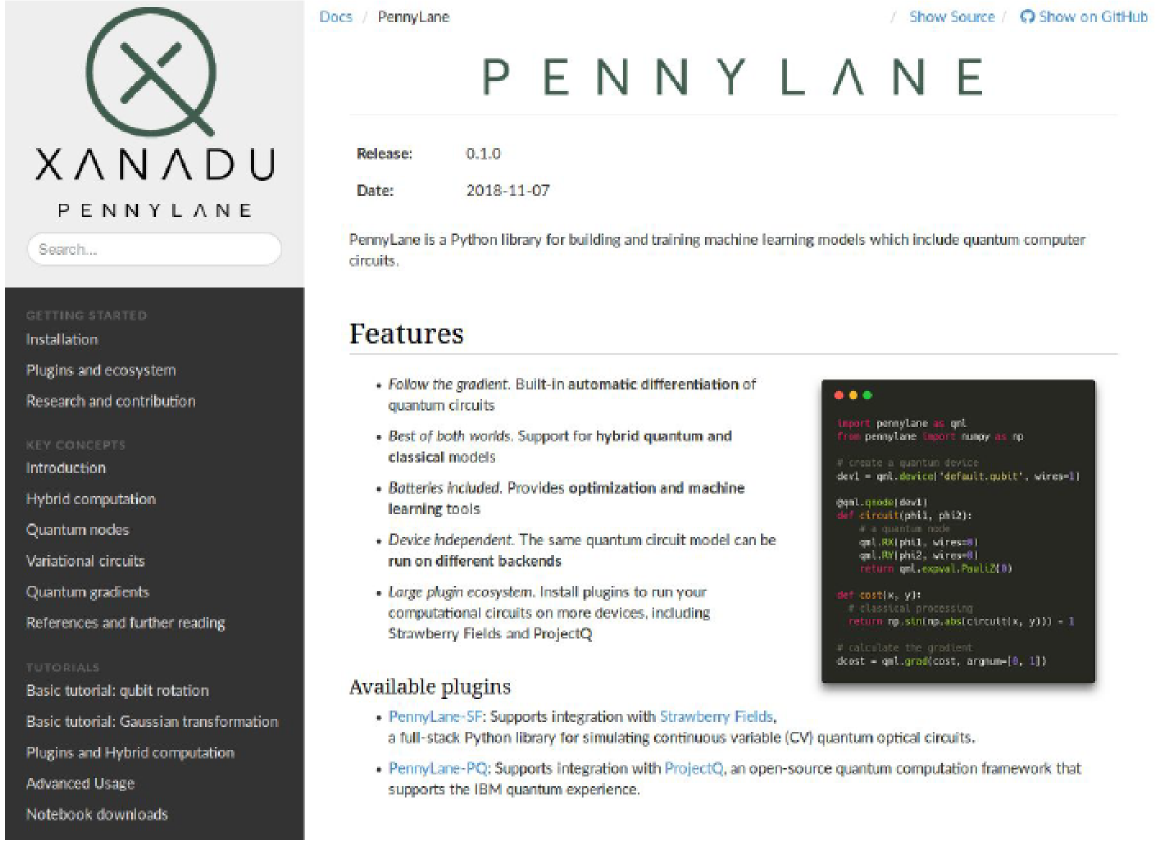
Hybrid Networks – Differentiable Computing



PennyLane

"The TensorFlow of quantum computing"

- Train a quantum computer the same way as a neural network
- Designed to scale as quantum computers grow in power
- Compatible with Xanadu, IBM, Rigetti, and Microsoft platforms



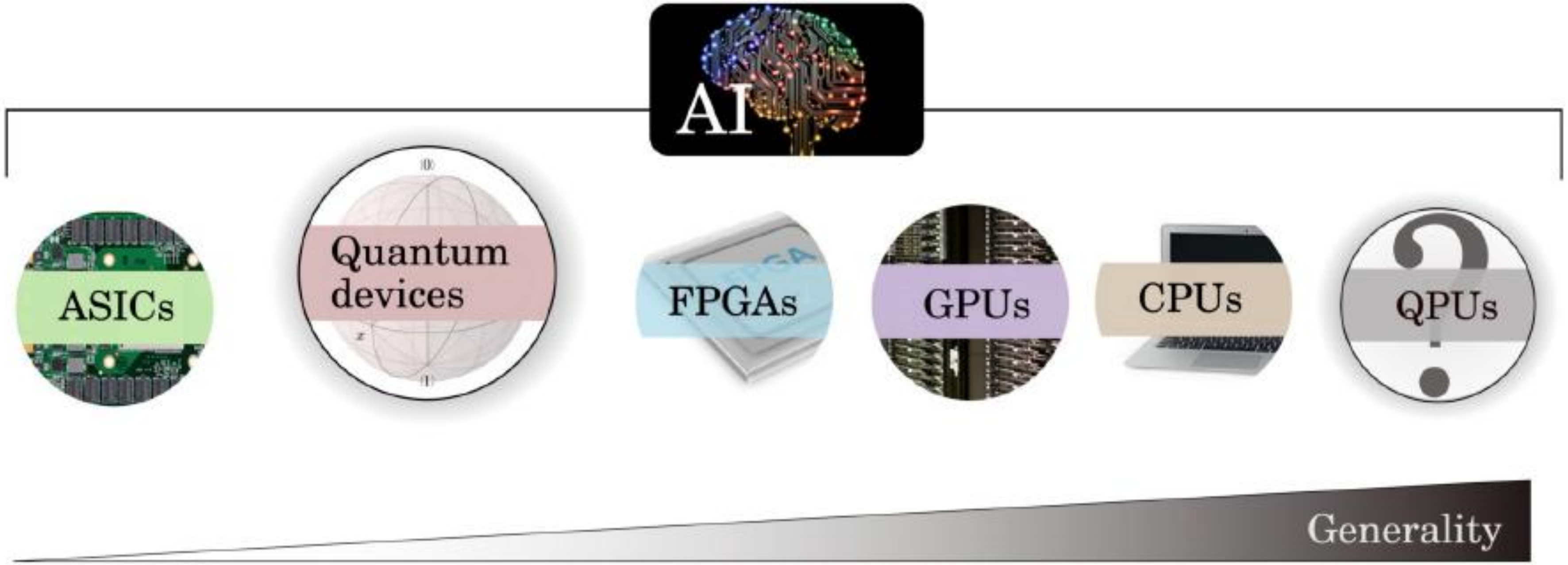
https://github.com/XanaduAI/pennylane

https://pennylane.ai

Other ecosystem

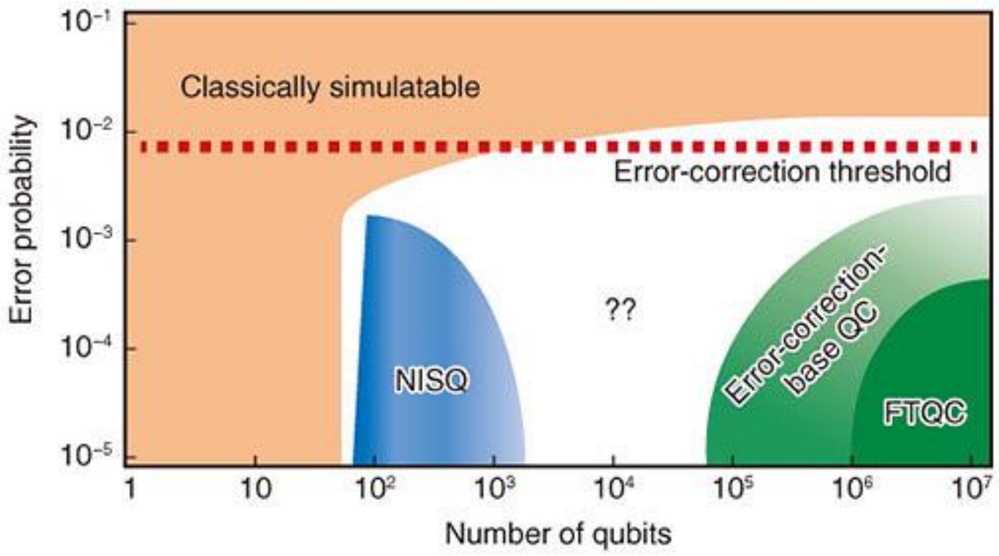
PENNYLANE
PyTorch
TensorFlow
STRAWBERRY FIELDS
rigetti Forest
Qiskit
NumPy
Microsoft Q#

Quantum Processing Units

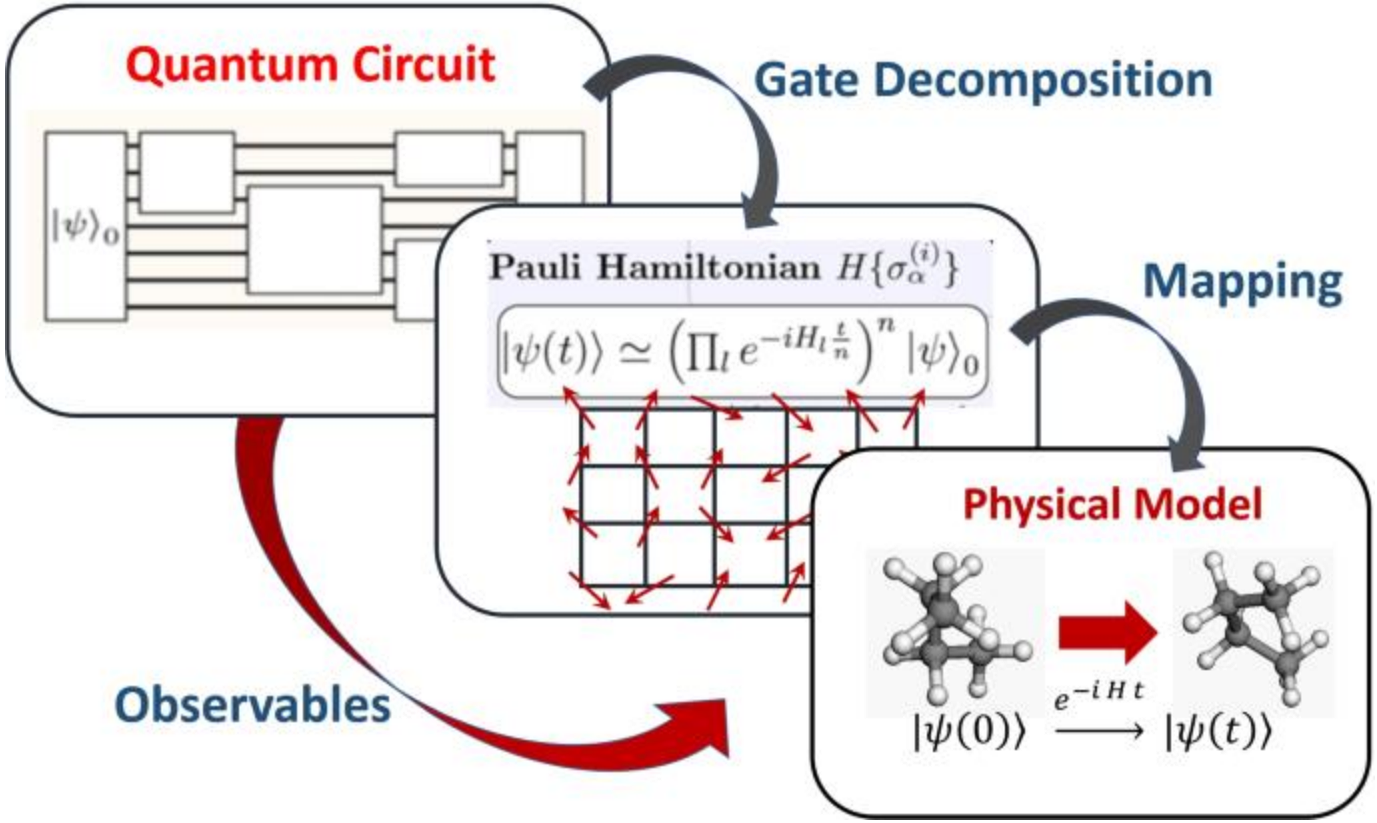


Noisy Intermediate Scale Quantum Era (NISQ)

- Upto 1000 qubits
- Suffers from decoherence
  - No error correction
- No fault tolerance



NISQ Processor Design

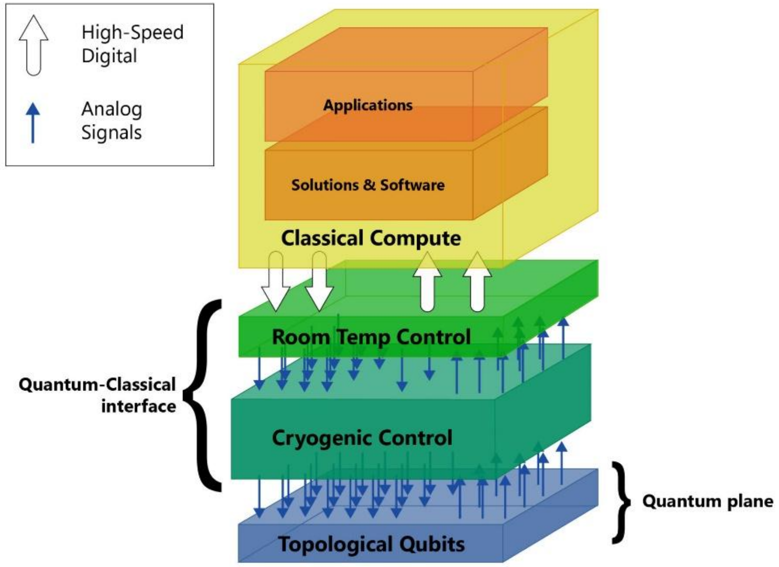


Quantum Processors

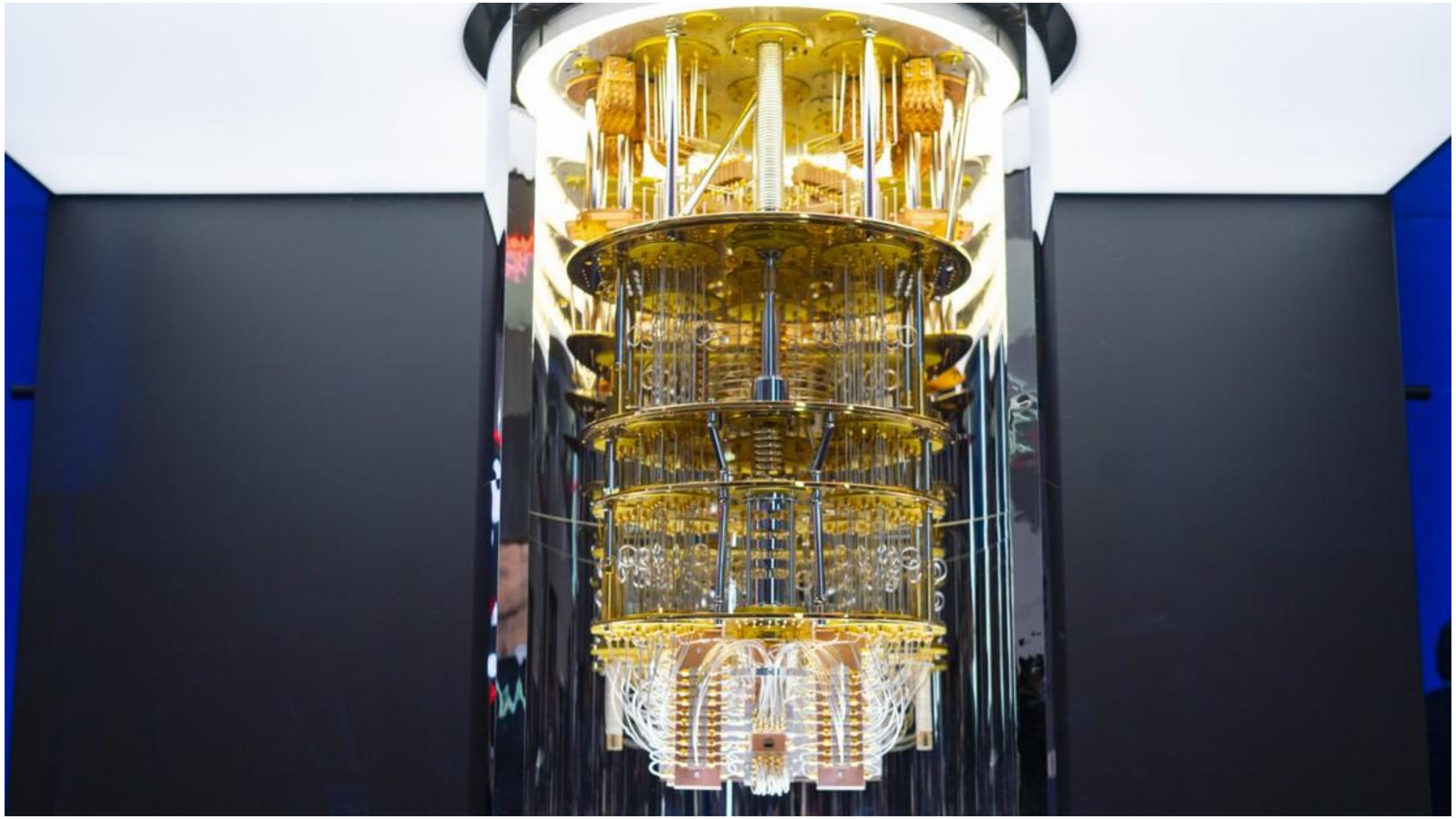
- Circuit based
- Annealing based
- Analog

- IBM
- Google
- Intel
- Riggeti
- Xanadu

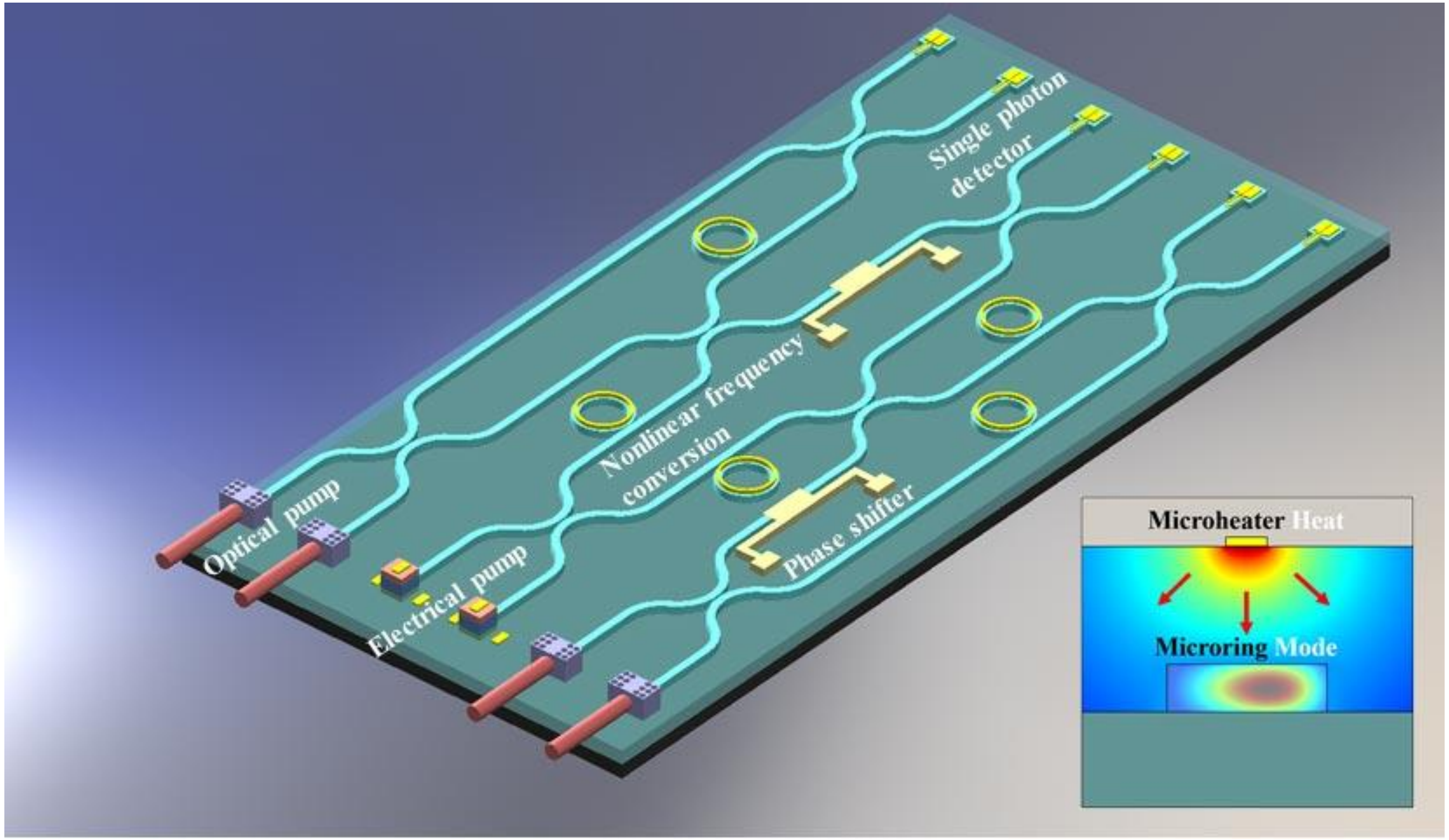
- Superconducting
- Photonics
- Semiconductor spin
- Trapped ion



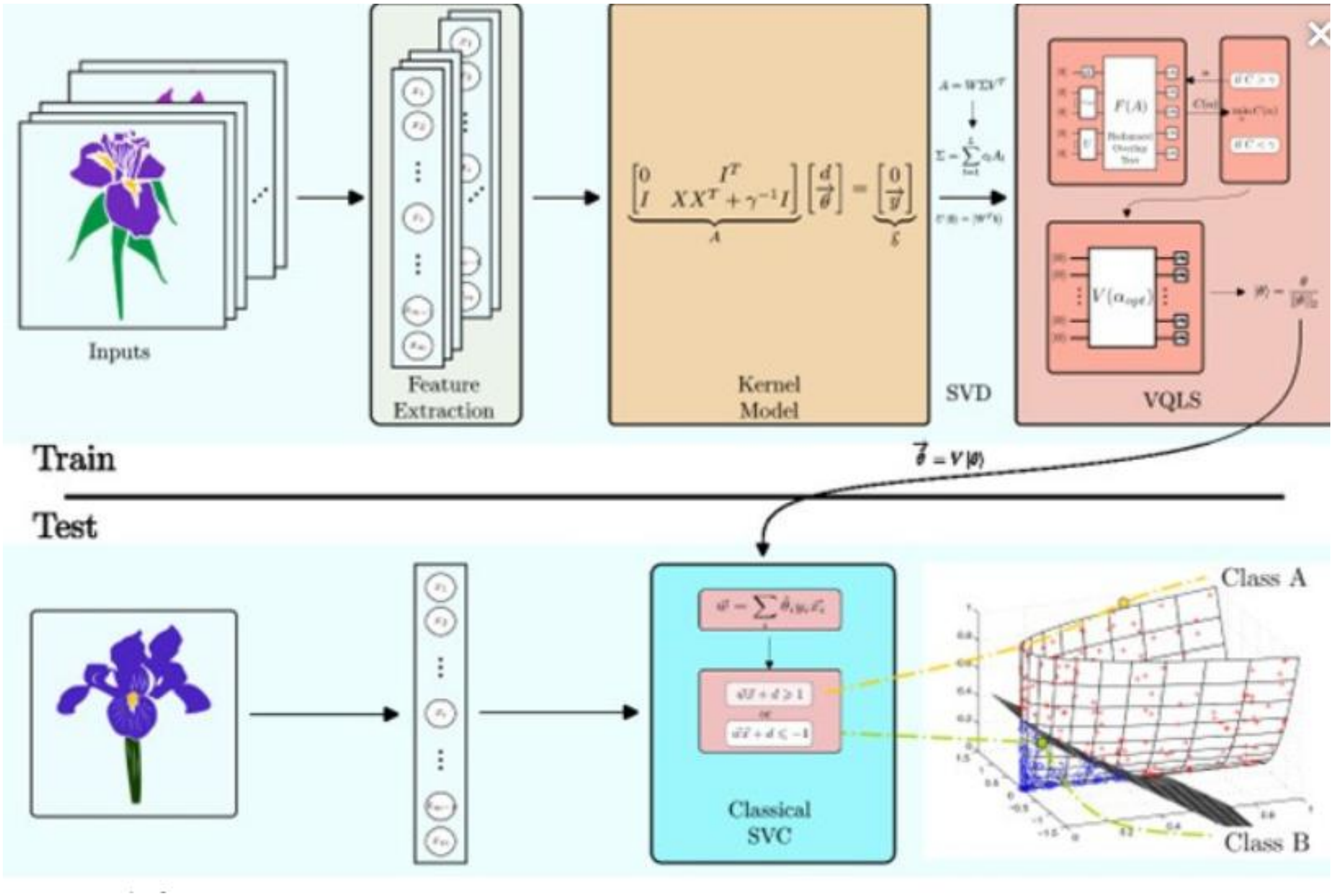
IBM Superconducting Quantum Processor



Photonic Quantum Processor



QNN for Plant Image Feature Extraction



EQID: Entangled quantum image descriptor an approach for early plant disease detection, Attri et al, Crop Protection, 2025

QNN for Plant Image Enhancement

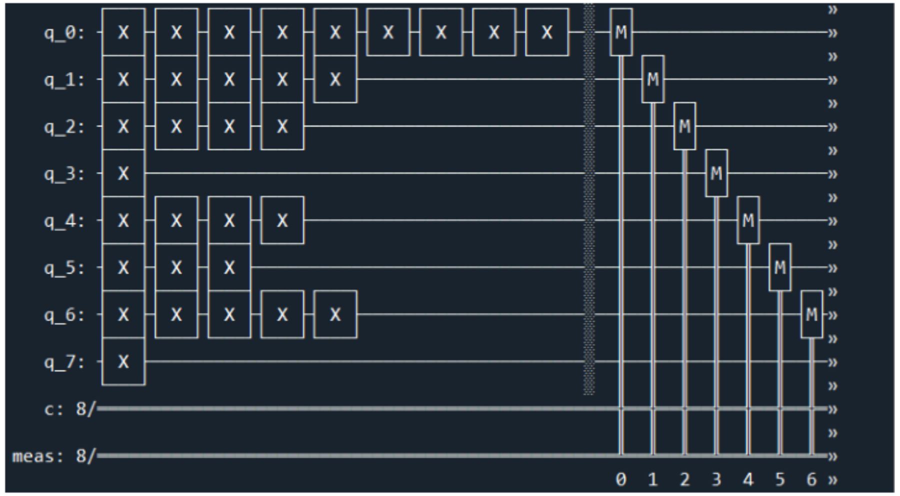
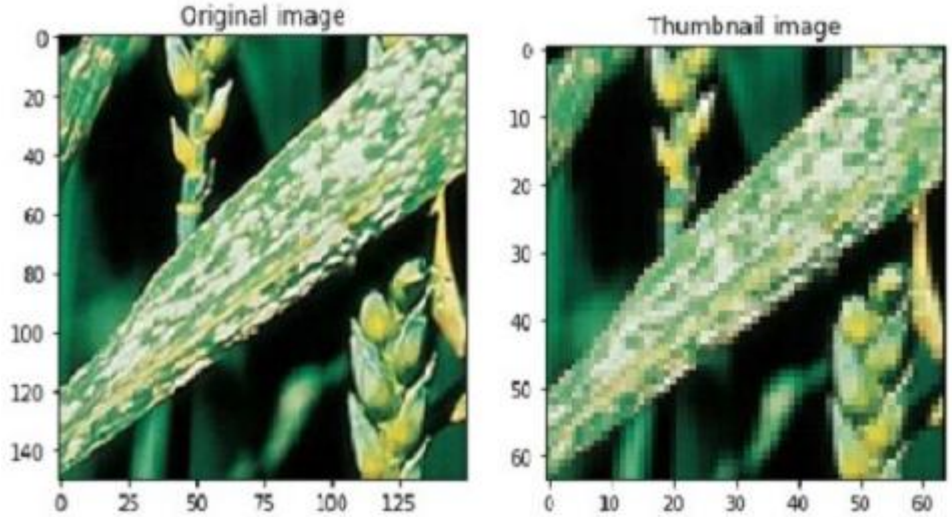
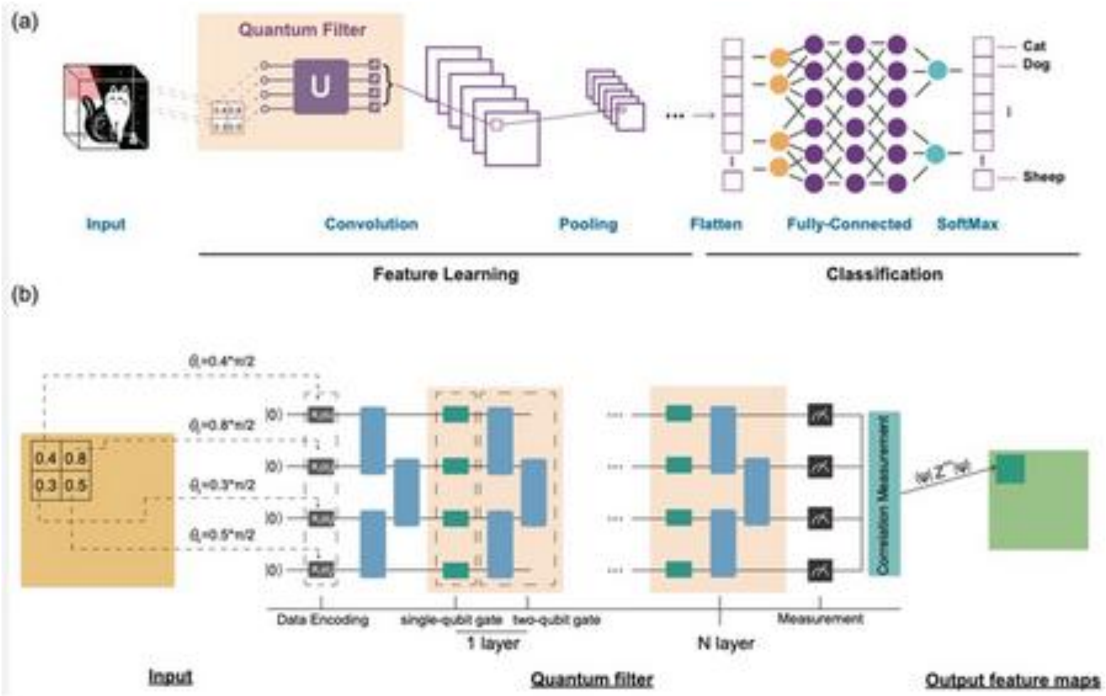


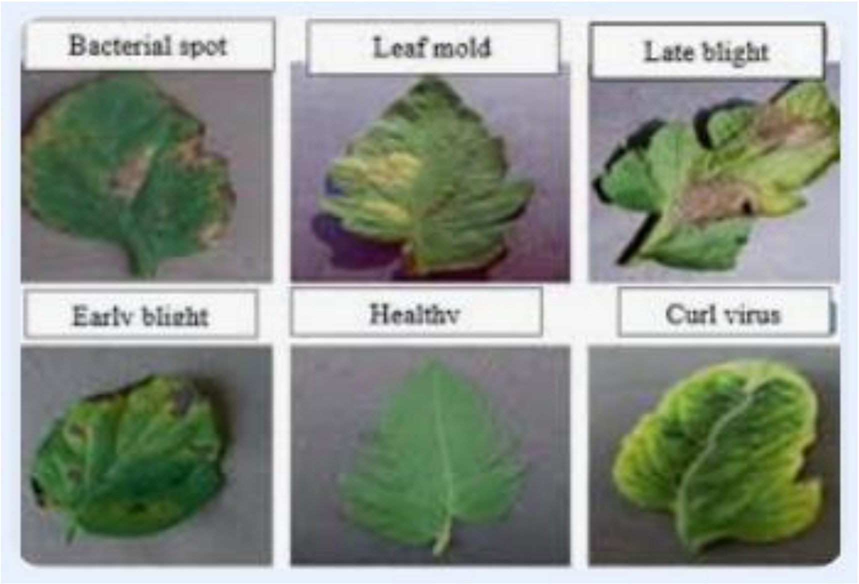
Fig. 6. Quantum contrast enhancement and noise filtering circuit.



Application of quantum computing in image processing for recognition of infectious diseases of wheat, Mukhadieveda et al (CIBTA-III-2024)

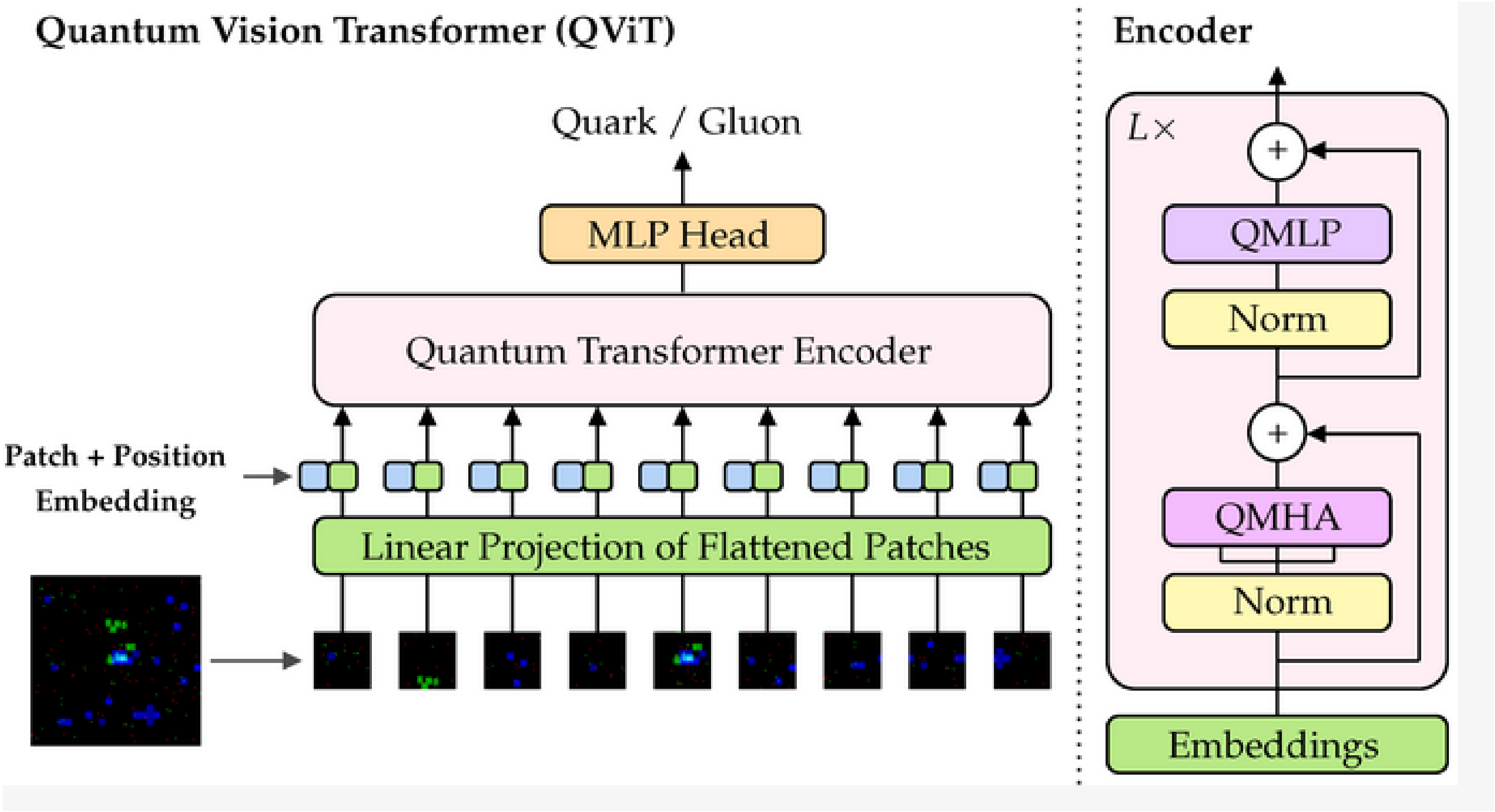
Quantum Convolutional Neural Network





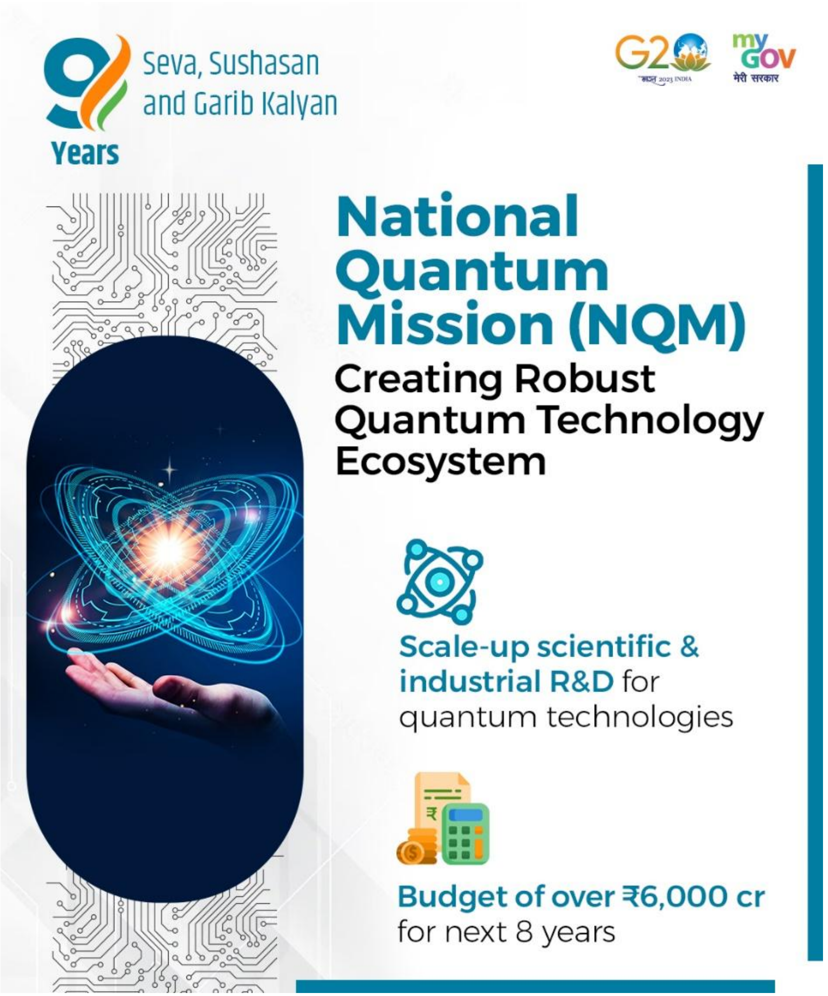
Quantum Convolutional Neural Network for Agricultural Mechanization and Plant Disease Detection, (ICIPCN 2023)

Quantum Vision Transformers



Cherrat et al Quantum Vision Transformers, Quantum, 2024

National Quantum Mission



To build a 1000 qubit quantum computer

- Computing
- Communication
- Sensing
- Material science

Thank you!