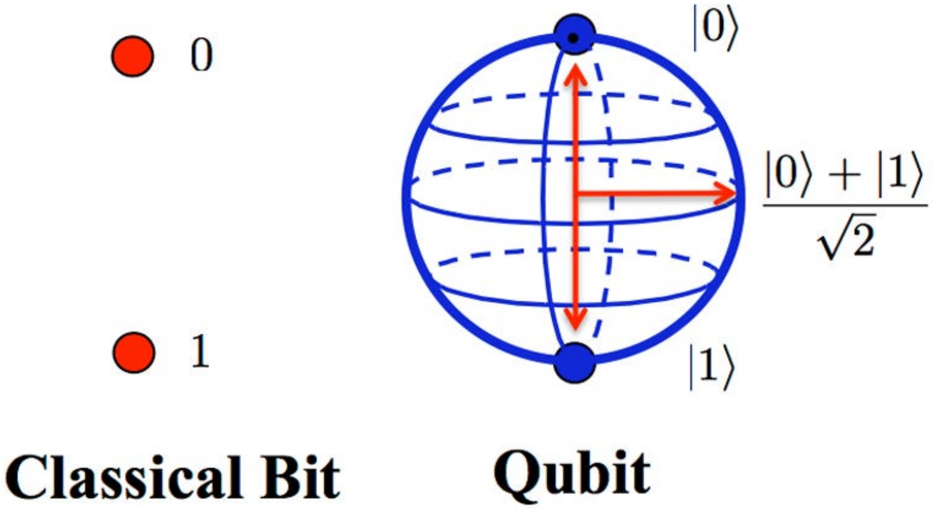
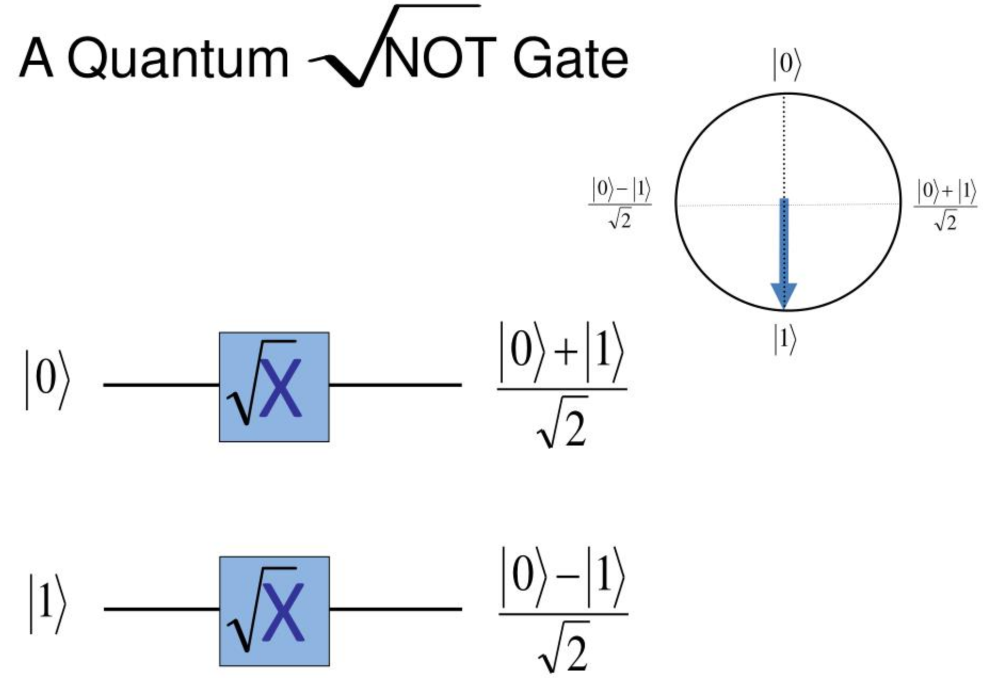
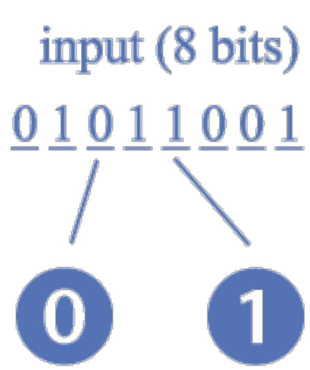
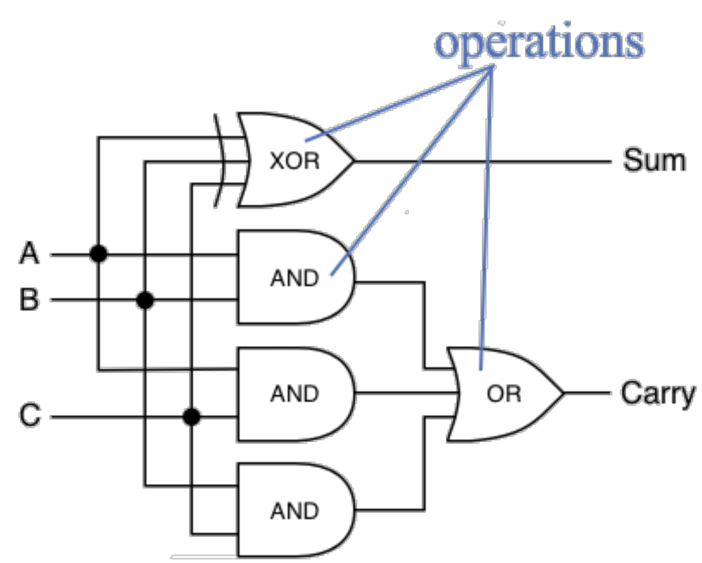
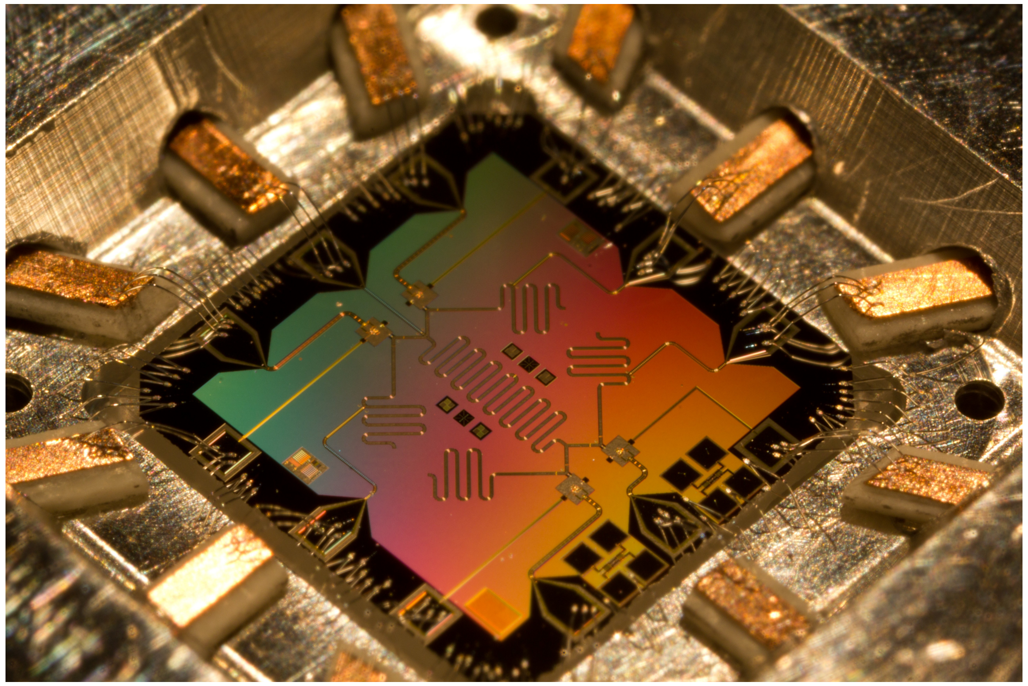




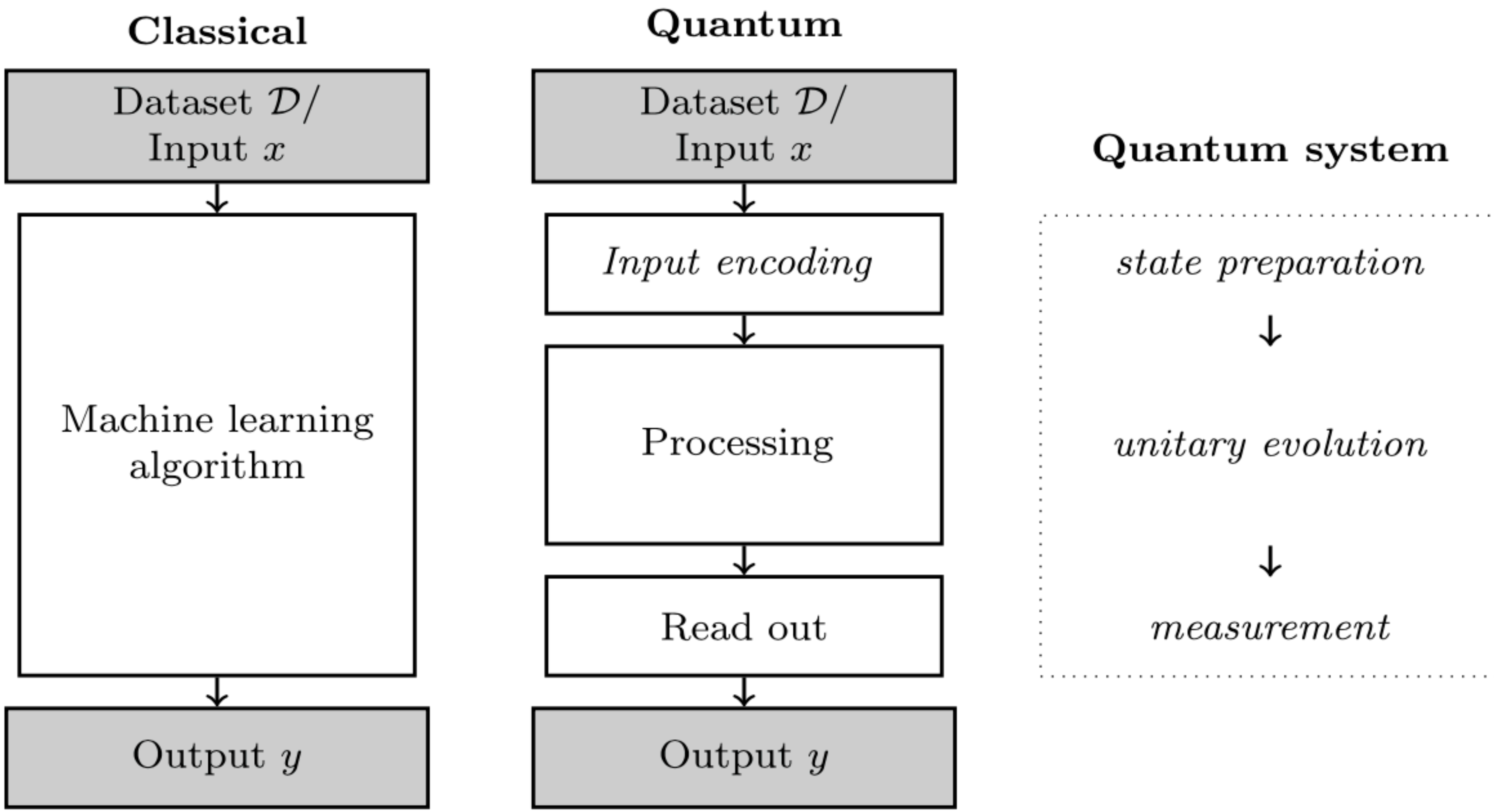
Quantum Gate Computing Basics

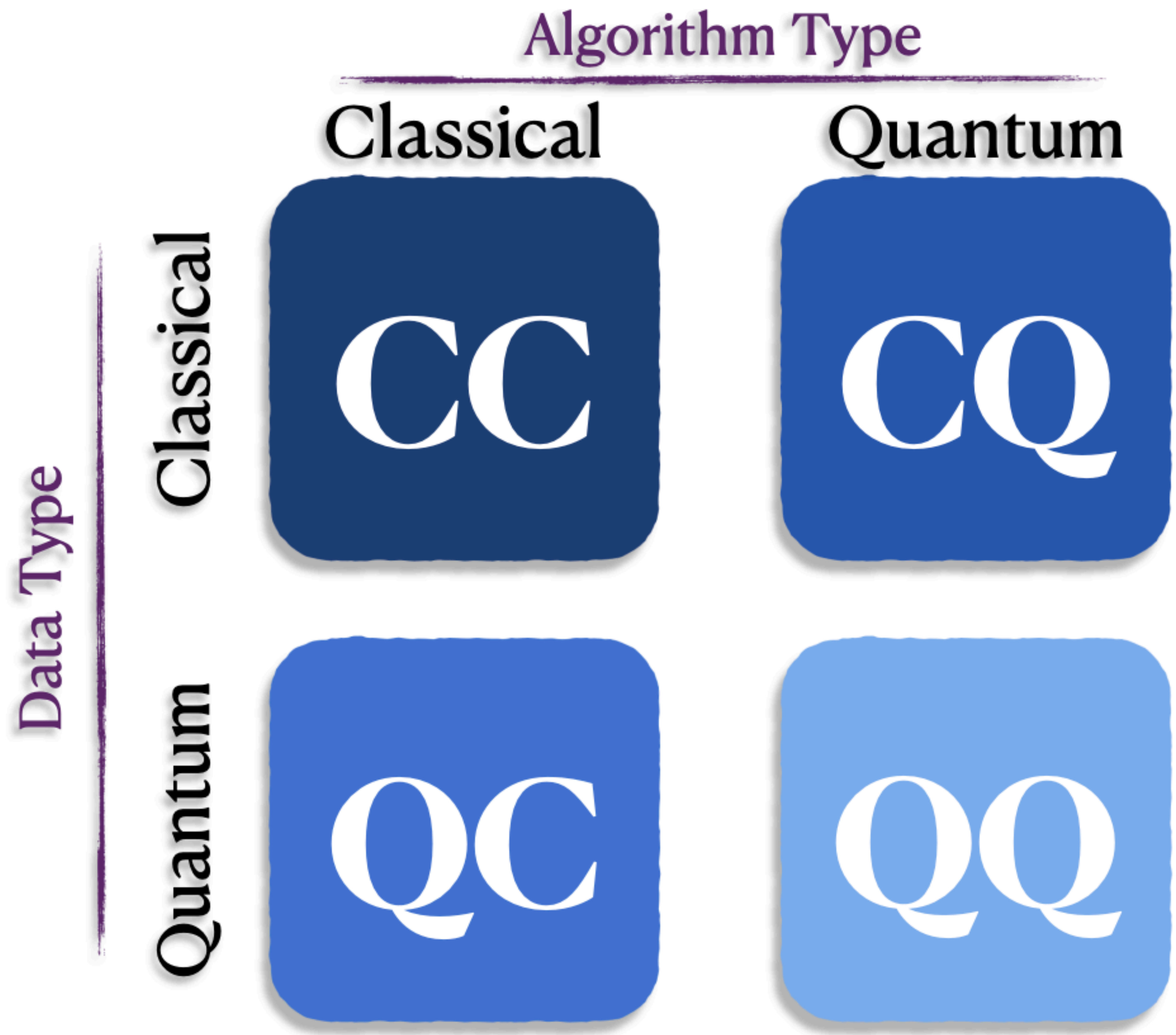


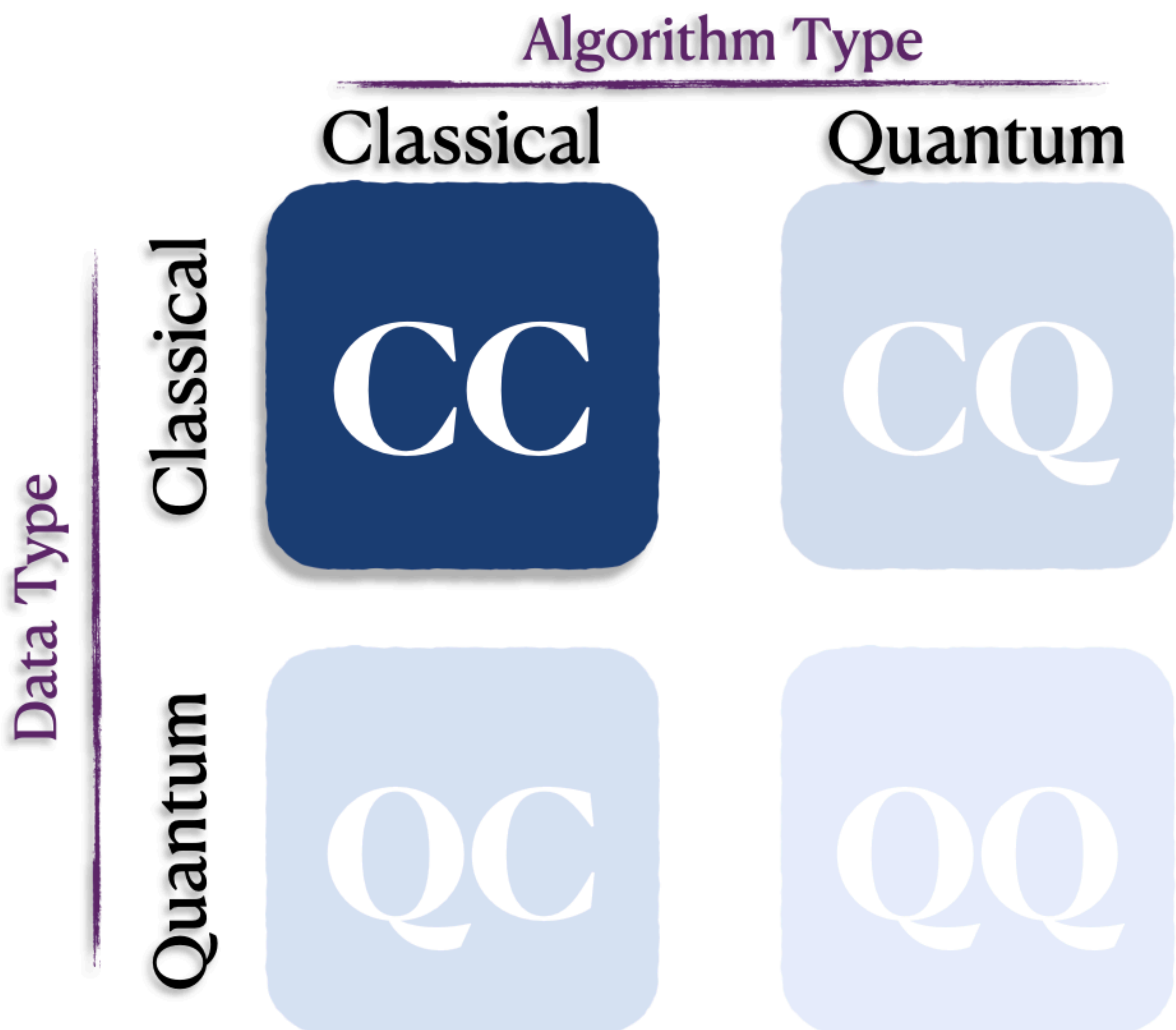




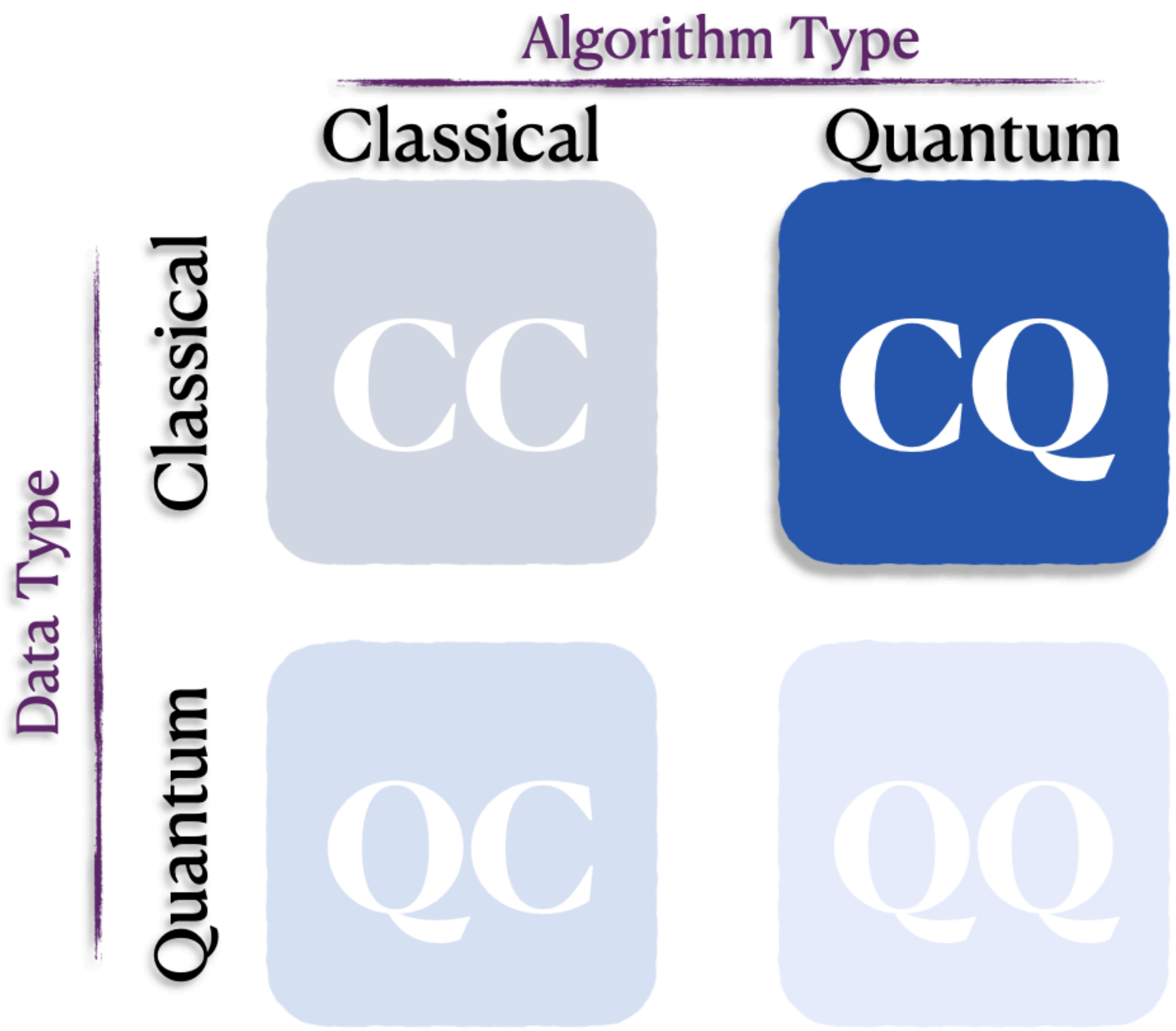
Structure of a Quantum Machine Learning Algorithm



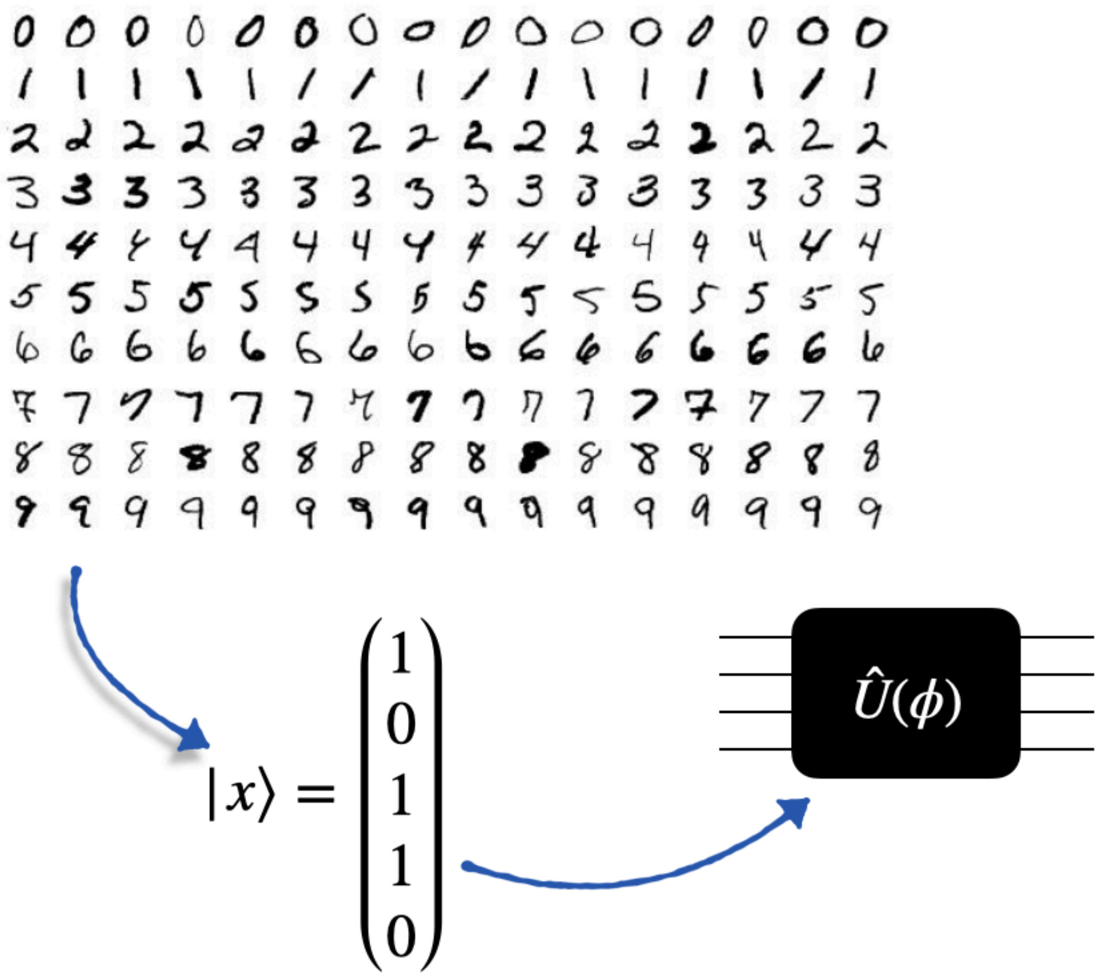

Algorithm Type
Classical
Quantum
Data Type
Classical
Quantum
CC
CQ
QC
QQ

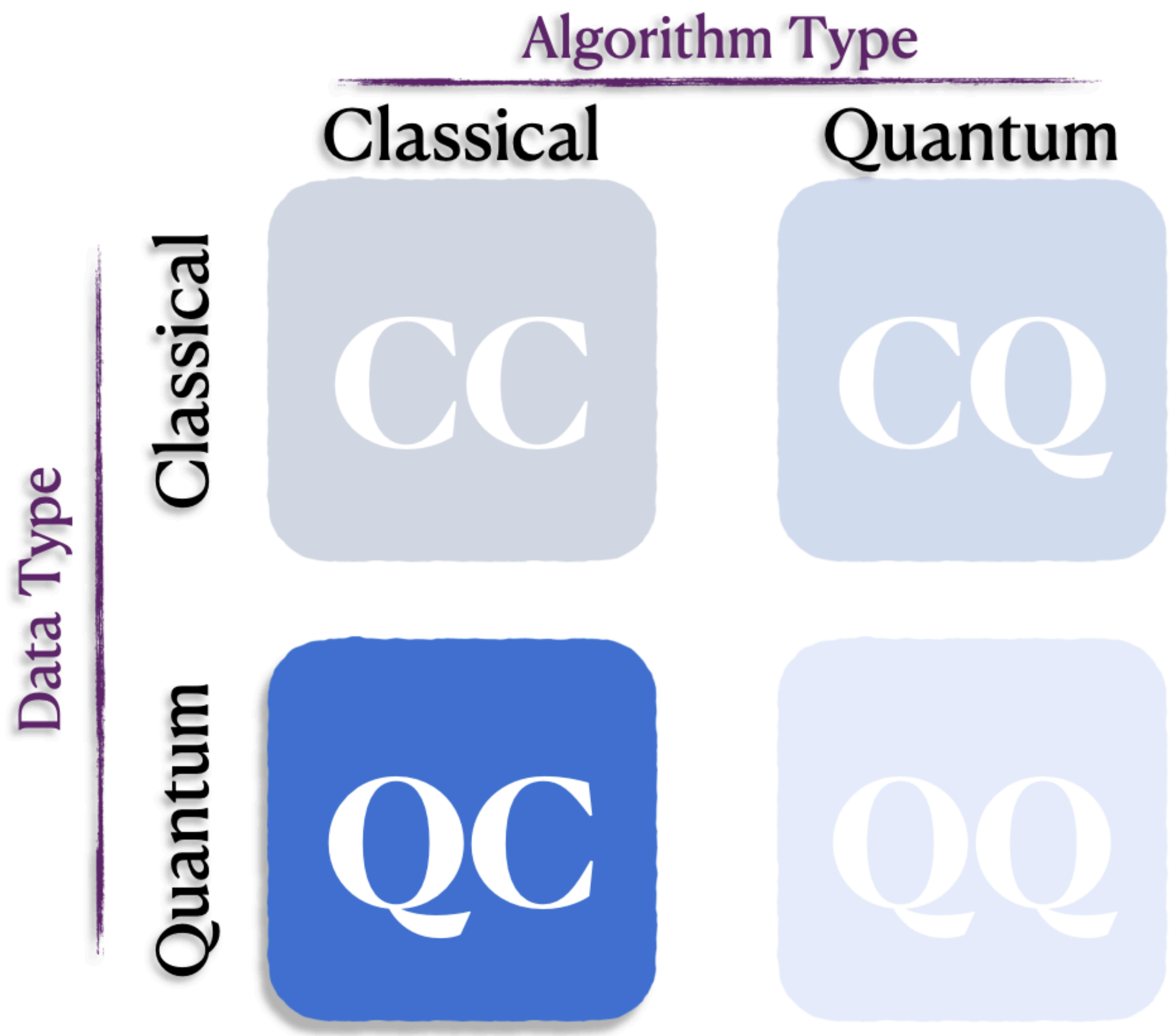


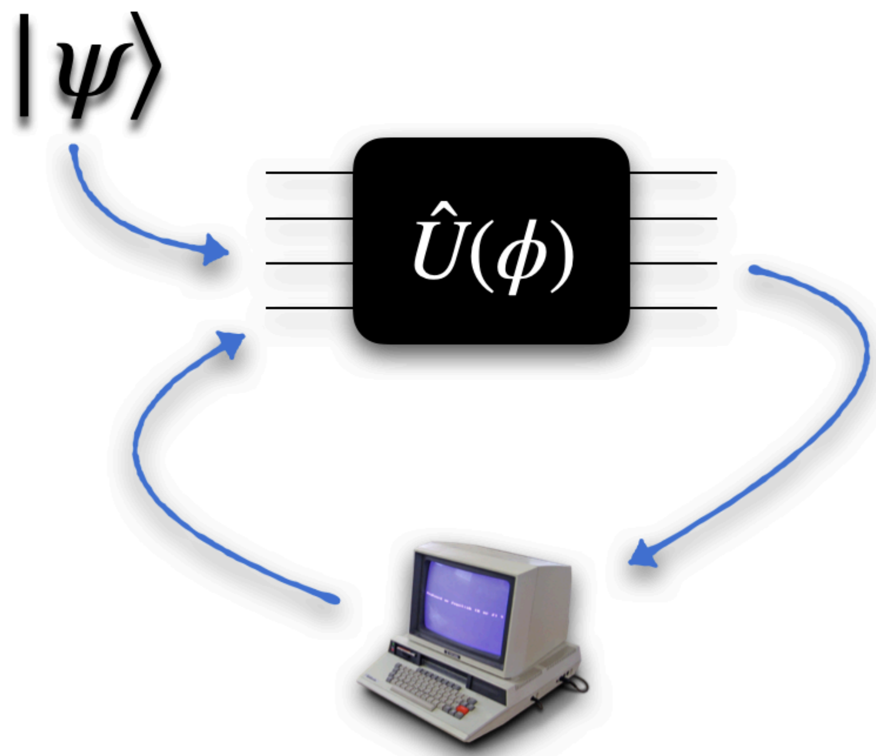
All the methods discussed in the lectures you had over the last 2 weeks



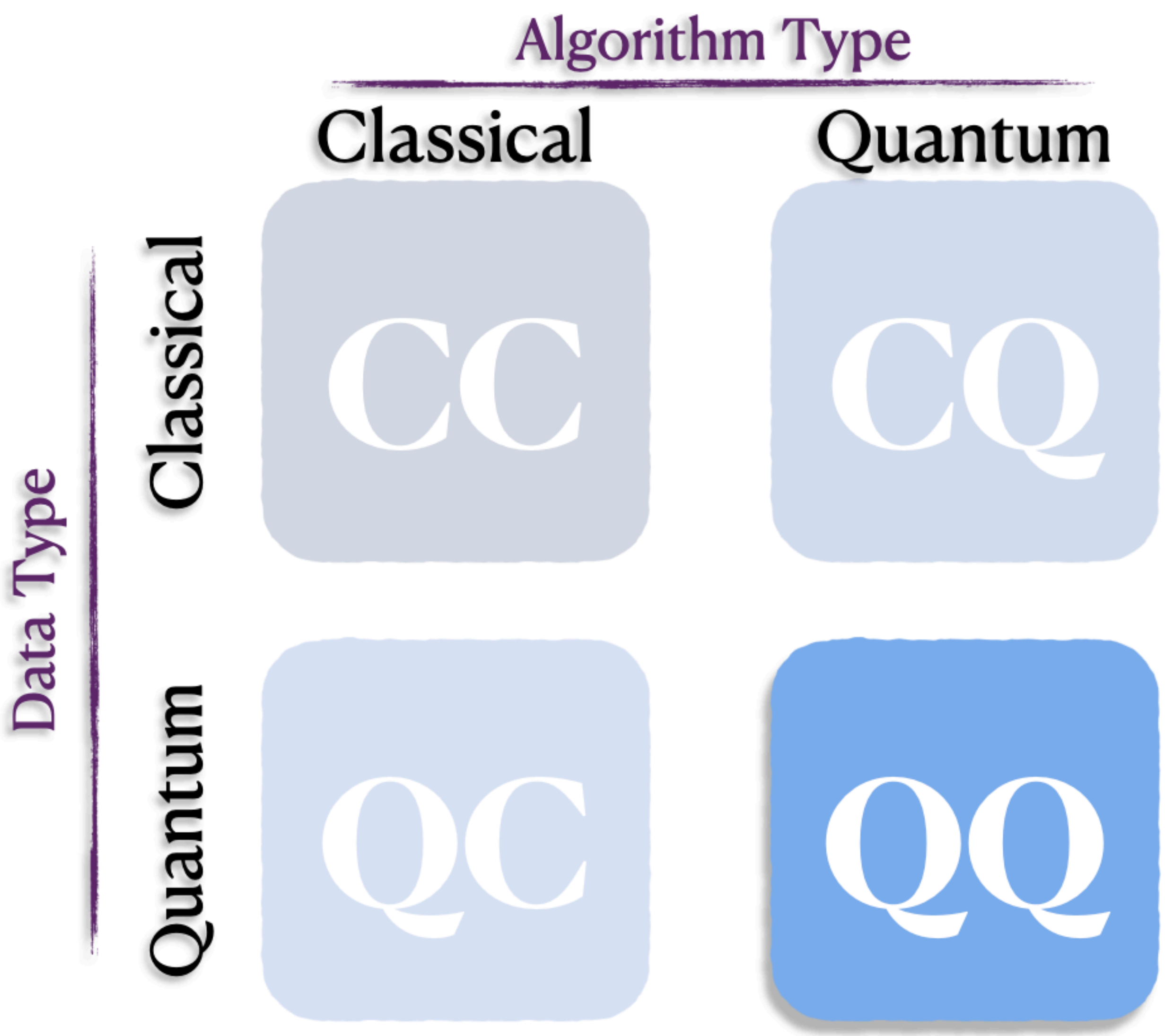
Classical data processed via quantum algorithms on quantum devices



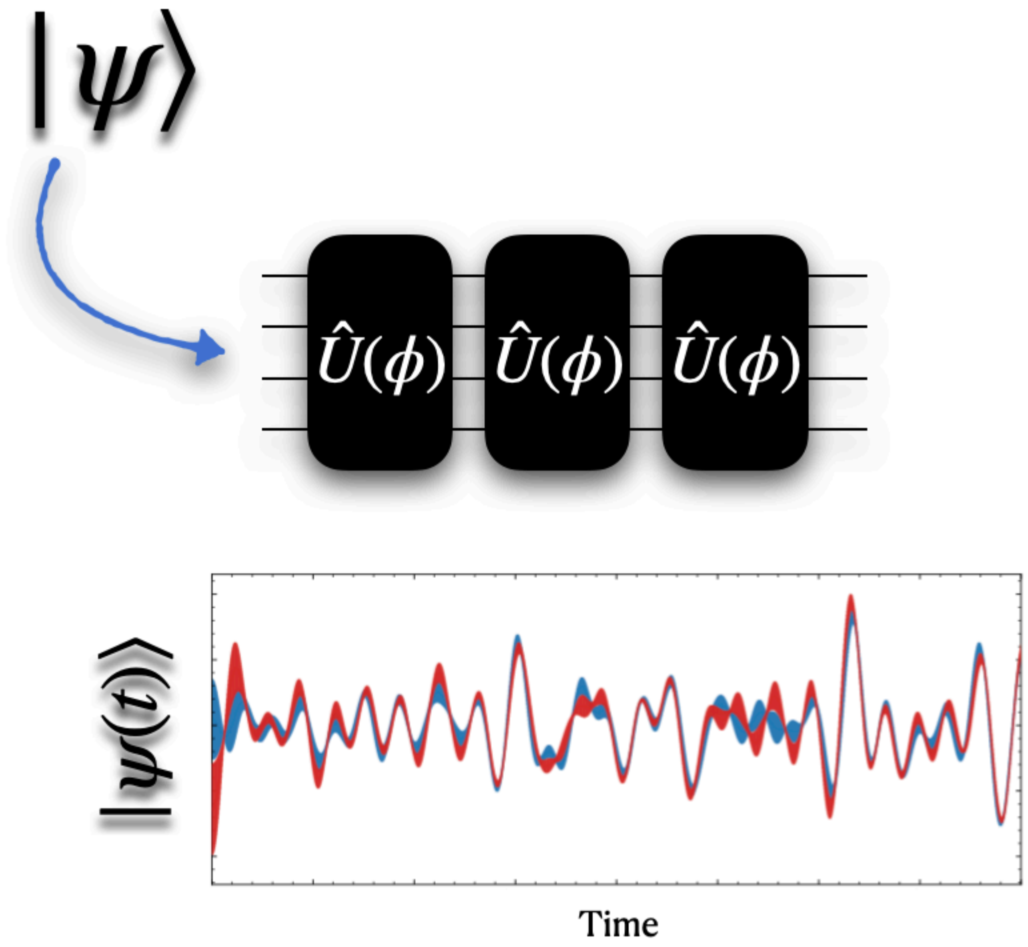




❖ Also Tensor Networks

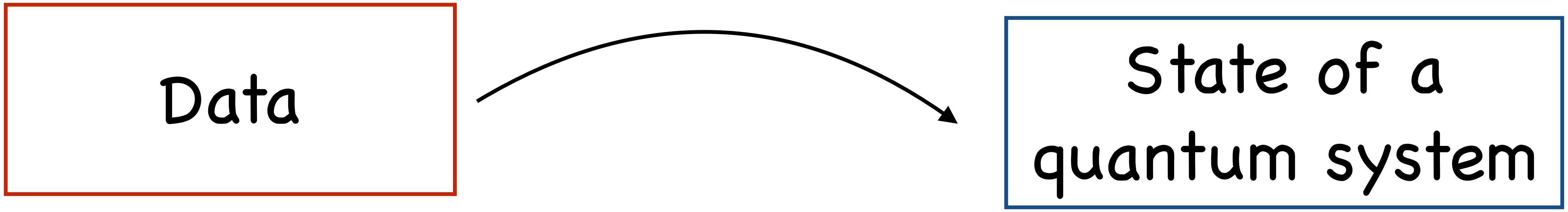


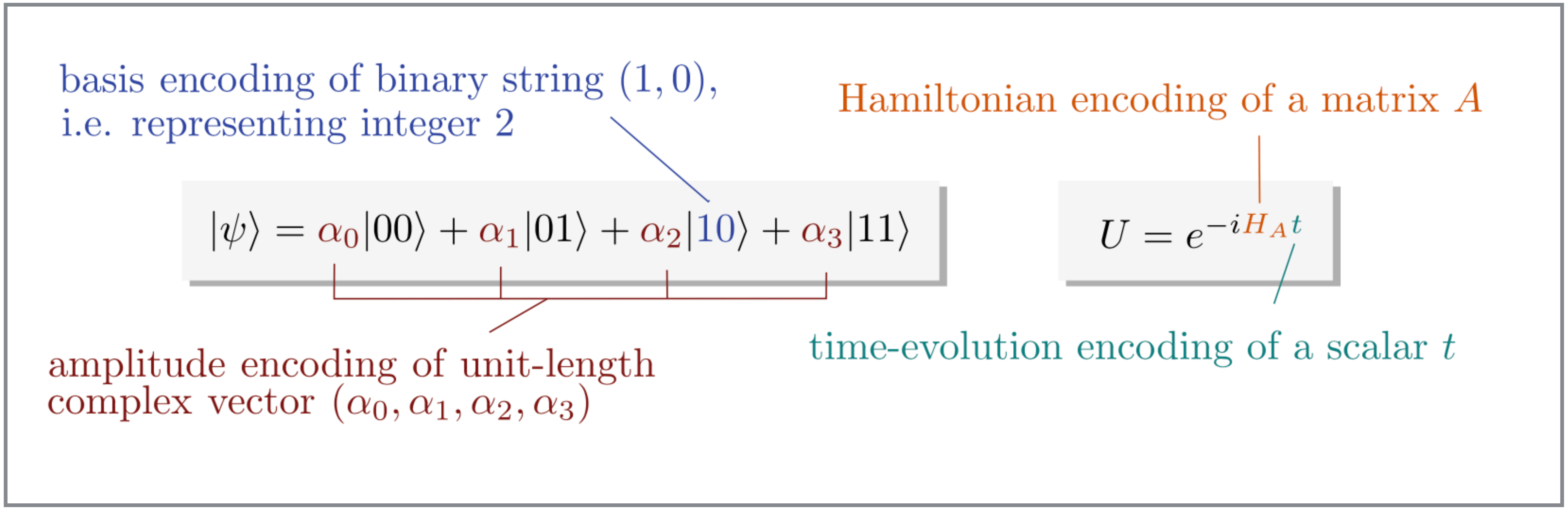
Algorithm Type
Classical
Quantum
Data Type
Classical
Quantum
CC
CQ
QC
QQ



$|\psi\rangle$
$\hat{U}(\phi)$
$\hat{U}(\phi)$
$\hat{U}(\phi)$
$|\psi(t)\rangle$
Time

Encoding





data encoding in different parts of the state and operator description

Basis encoding:

maps a collection of items into the states forming an orthonormal basis of the Hilbert space of the considered quantum system.

The orthonormal basis $\{|x\rangle\}_{x \in X}$, called **computational basis**, is made by the eigenstates of a reference observable measured on the considered quantum system. For instance, a bit can be encoded into a qubit by the mapping $0 \to |0\rangle$, $1 \to |1\rangle$. Then the n-bit strings $(x_1 \cdots x_n)$ can be encoded into the states of n qubits forming an orthonormal basis of a 2^n-dimensional Hilbert space H_n:

$$\mathbb{B}^n \ni (x_1 \cdots x_n) \mapsto |x_1 \cdots x_n\rangle \in \mathsf{H}_n$$

$\longrightarrow$ can prepare superposition of data that can be processed in parallel, e.g.

$$|\psi\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle$$

Binary encoding into basis states

Represent numbers as binaries, each binary digit requires a qubit

basis vector coefficient {0,1}

$$x = \sum_{k=1}^{\tau-1} b_k \frac{1}{2^k}$$

binary fraction rep.

data vector

sign

quantum state

$$x = (0.1, -0.6, 1.0) \longrightarrow \begin{matrix} 0.1 \rightarrow 0\ 0001\ldots \\ -0.6 \rightarrow 1\ 1001\ldots \\ 1.0 \rightarrow 0\ 1111\ldots. \end{matrix} \longrightarrow |00001\ 11001\ 01111\rangle$$

Time-Evolution Encoding

Associates the input data value x with the time evolution parameter t

$$U(x) = e^{-ixH}$$

In quantum machine learning, this kind of encoding is particularly popular when encoding classical trainable parameters into a quantum circuit. The most common choice are the Pauli rotation gates, in which $H = \frac{1}{2}\sigma_a$ and a ∈ {x, y, z}. Successive gates or evolutions of the form U(x) can be used to encode a real-valued vector **x** ∈ R^N

data	Encoded with RY gate applied to initial state |0>
-0.438	$\|\psi(-0.438)\rangle = \cos(-0.438/2)\|0\rangle + \sin(-0.438/2)\|1\rangle$
	$\approx 0.976\|0\rangle - 0.217\|1\rangle$

sine/cosine structure typical for Time-Evolution Encoding -> leads to Fourier-type dependence of amplitudes on the inputs

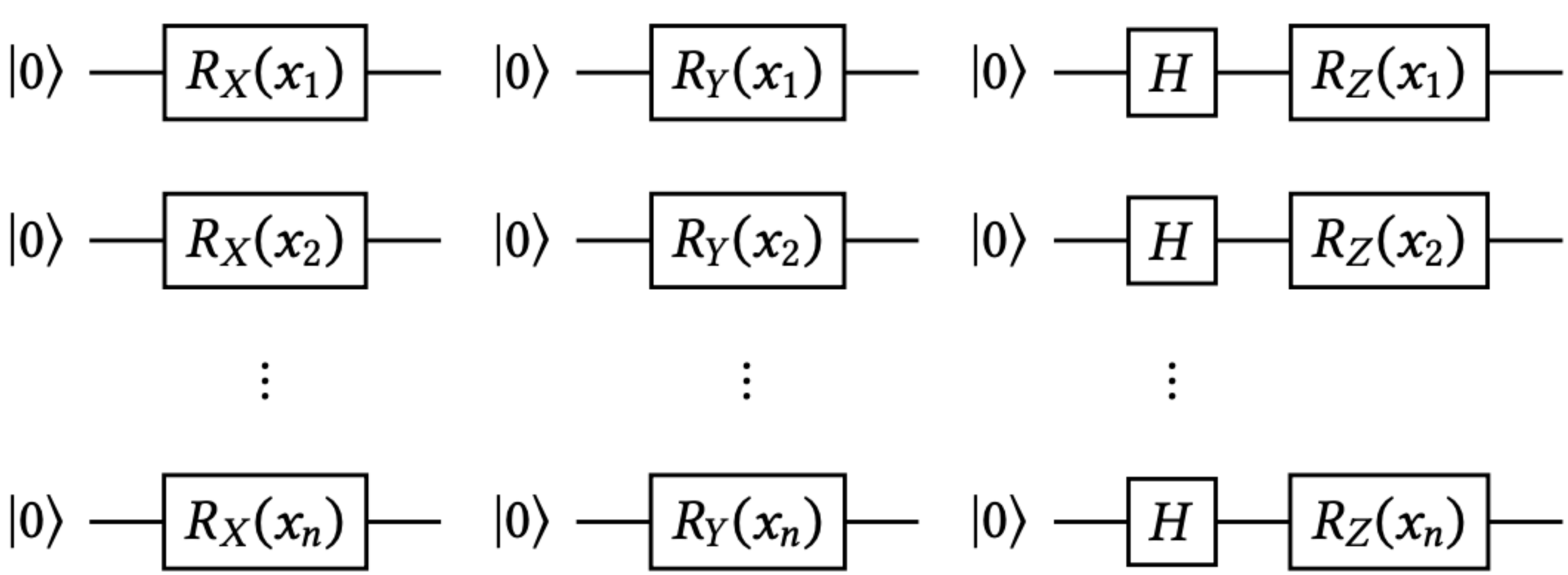
Angle/Rotation encoding

When used on an n-qubit circuit, this feature map of angle encoding can take up to n numerical inputs $x_1, \ldots, x_n$. The action of its circuit consists in the application of a rotation gate on each qubit j parametrised by the value x_j. In this feature map, we are using the x_j values as angles in the rotations, hence the name of the encoding.

Example

as RZ|0> doesnt do anything

x normalised [0,2pi)



In example of simple Pauli-X rotation, transforms real-valued N-dimensional input vector $\mathbf{x} \in \mathbb{R}^N$ as

$$\phi_1(\mathbf{x}) = \begin{pmatrix} \sin(x_1)\sin(x_2)\dots\sin(x_N) \\ \sin(x_1)\sin(x_2)\dots\cos(x_N) \\ \vdots \\ \cos(x_1)\cos(x_2)\dots\sin(x_N) \\ \cos(x_1)\cos(x_2)\dots\cos(x_N) \end{pmatrix}$$

(need N qubits)

encoding can be repeated multiple times, e.g.

$$\phi_1(\mathbf{x}) \otimes \cdots \otimes \phi_1(\mathbf{x}) = \begin{pmatrix} x_1 x_1 \dots x_1 \\ x_1 x_1 \dots x_2 \\ \vdots \\ x_N x_N \dots x_N \end{pmatrix}$$

⟶ results in non-linearities and higher expressivity of model

⟶ repeated encoding used to show universal approximation theorem for variational quantum circuits

Hamiltonian Encoding:

For some applications, it can be useful to encode matrices into the Hamiltonian of a time evolution. The basic idea is to associate a Hamiltonian H with a square matrix **A**. In case **A** is not Hermitian, one can sometimes use the trick of encoding

$$H_{\mathbf{A}} = \begin{pmatrix} 0 & \mathbf{A} \\ \mathbf{A}^\dagger & 0 \end{pmatrix}$$

instead, and to perform the computations in two subspaces of the Hilbert space. Hamiltonian encoding allows us to extract and process the eigenvalues of **A**, for example, to multiply **A** or $\mathbf{A}^{-1}$ with an amplitude-encoded vector.

Amplitude encoding:

Represent classical data as amplitudes of a quantum state

$$|\psi_{\mathbf{x}}\rangle = \sum_{i=1}^{d} x_i |\phi_i\rangle \in \mathsf{H}$$

or for composite systems with $\sum_{ij} |a_{ij}|^2 = 1$

$$|\psi_A\rangle = \sum_{i,j=1}^{d} a_{ij} |\phi_i\rangle \otimes |\phi_j\rangle \in \mathsf{H} \otimes \mathsf{H}$$

Example:

data vector

$\mathbf{x} = (0.1, -0.6, 1.0)$ ⟶

normalised and padded data vector

$\mathbf{x} = (0.073, -0.438, 0.730, 0.000)$

↓

quantum state

$$|\psi_{\mathbf{x}}\rangle = 0.073|00\rangle - 0.438|01\rangle + 0.730|10\rangle + 0|11\rangle$$

This could also be encoded as a matrix A

$$\mathbf{A} = \begin{pmatrix} 0.073 & -0.438 \\ 0.730 & 0.000 \end{pmatrix}$$

Amplitude encoding uses much less qubits than basis encoding, however, routines to prepare amplitude vectors can be costly

Qsample encoding:

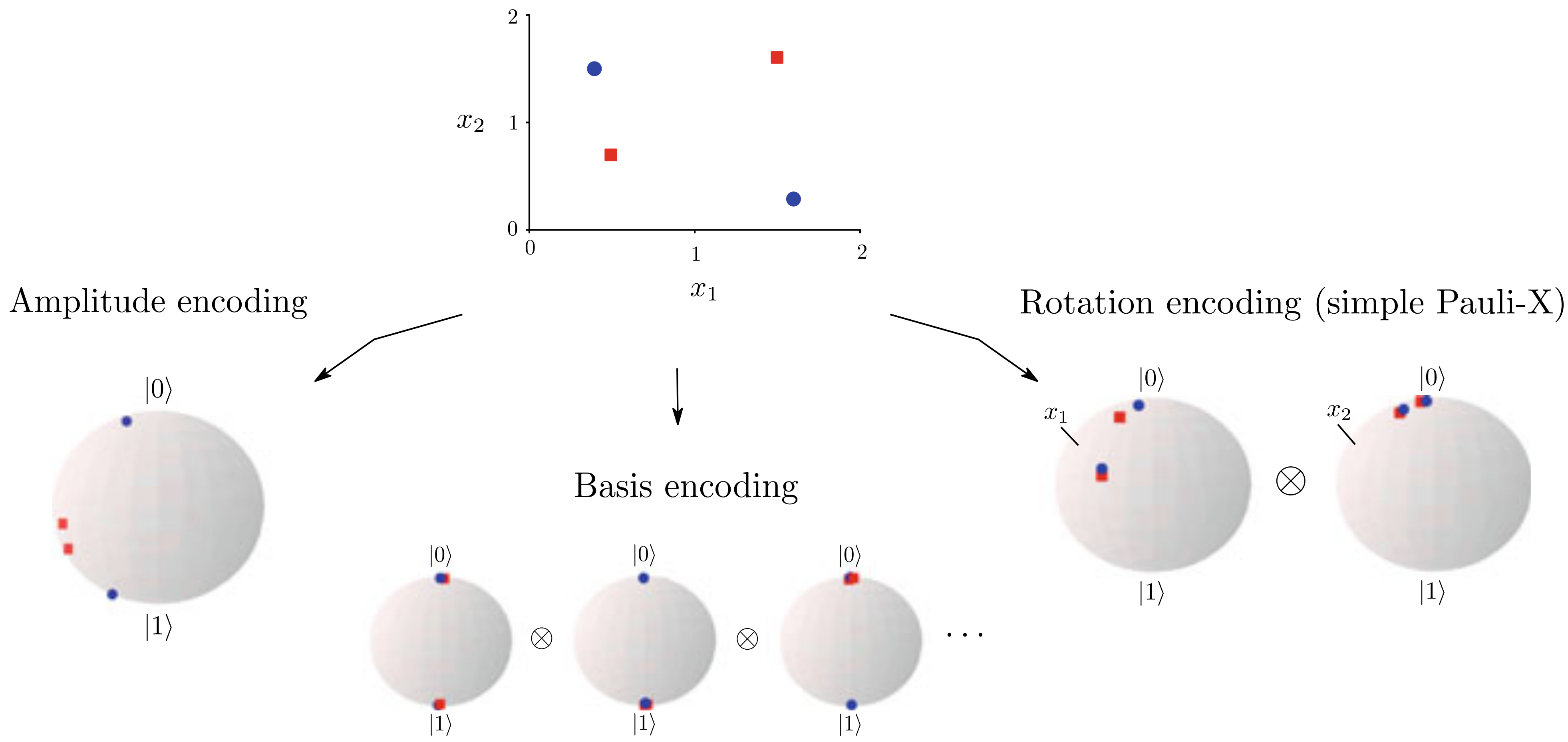
Given a probability distribution p on the finite set X, it can be encoded in the state:

$$|\psi_p\rangle = \sum_{x \in X} \sqrt{p(x)} |x\rangle \in \mathrm{H}$$

Repeated measurements on the state $|\psi_p\rangle$ with respect to the computational basis allow to sample the distribution p.

In a sense a hybrid case of basis and amplitude encoding since the information is represented by amplitudes, but the features are encoded in the qubits.

visualisation of data encoding

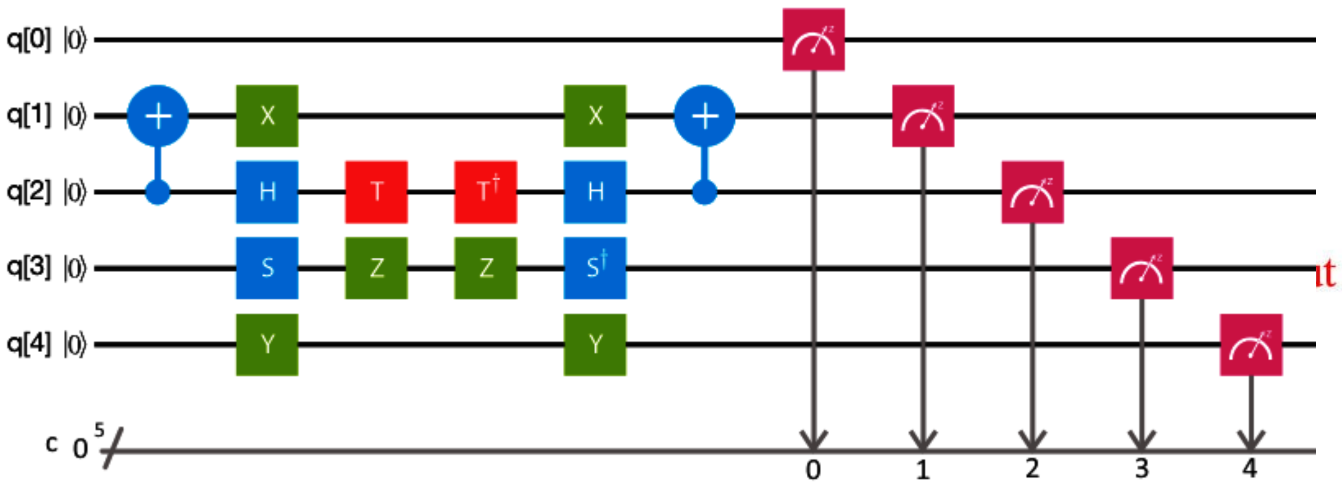


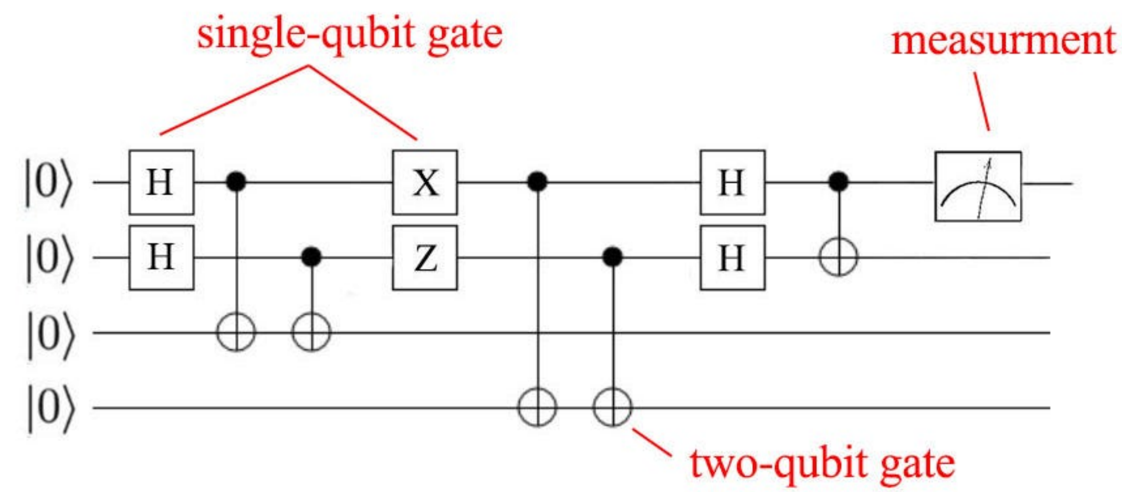
Encoding	# qubits	Runtime	Input type
Basis	$N\tau$	$\mathcal{O}(N\tau)$	Single input (binary)
Amplitude	$\log N$	$\mathcal{O}(N)$/ $\mathcal{O}(\log(N))$[a]	Single input
Angle	N	$\mathcal{O}(N)$	Single input
Hamiltonian	$\log N$	$\mathcal{O}(MN)$/ $\mathcal{O}(\log(MN))$[a]	Entire dataset

[a]Only applies under strict assumptions. see Schuld & Petruccione

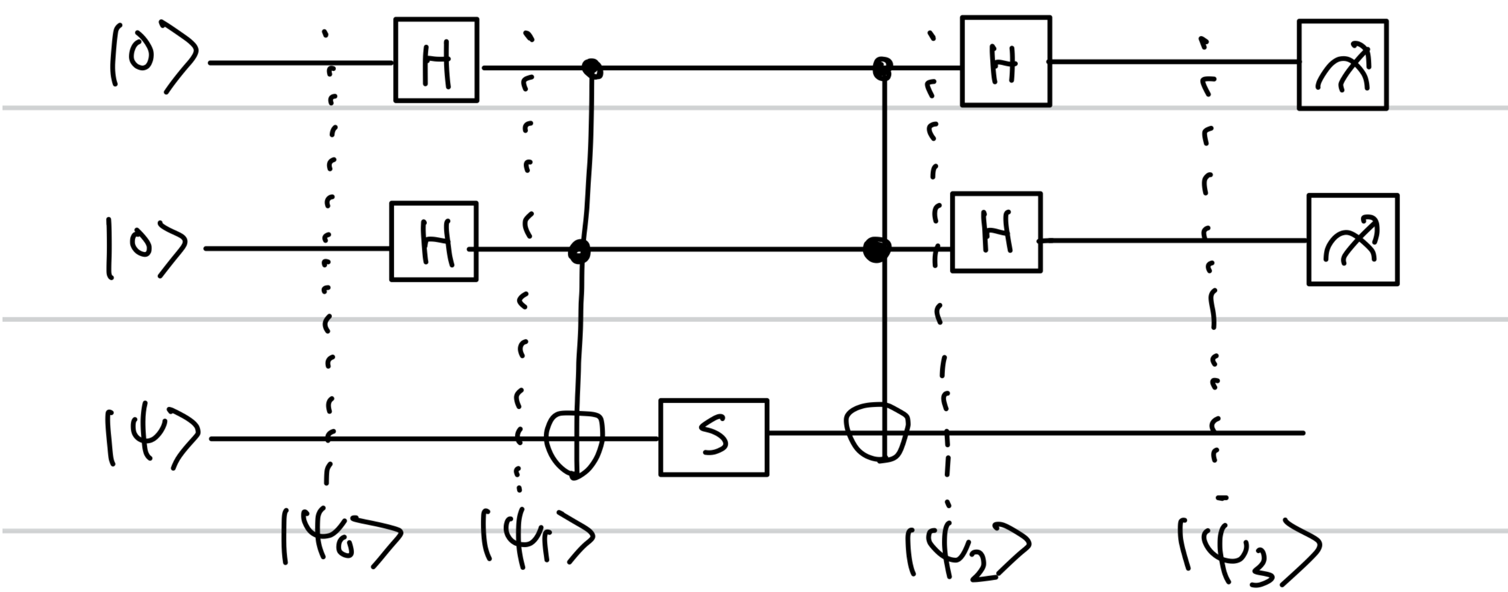
Encoding can be important for runtime of algo - crucial aspect of QC

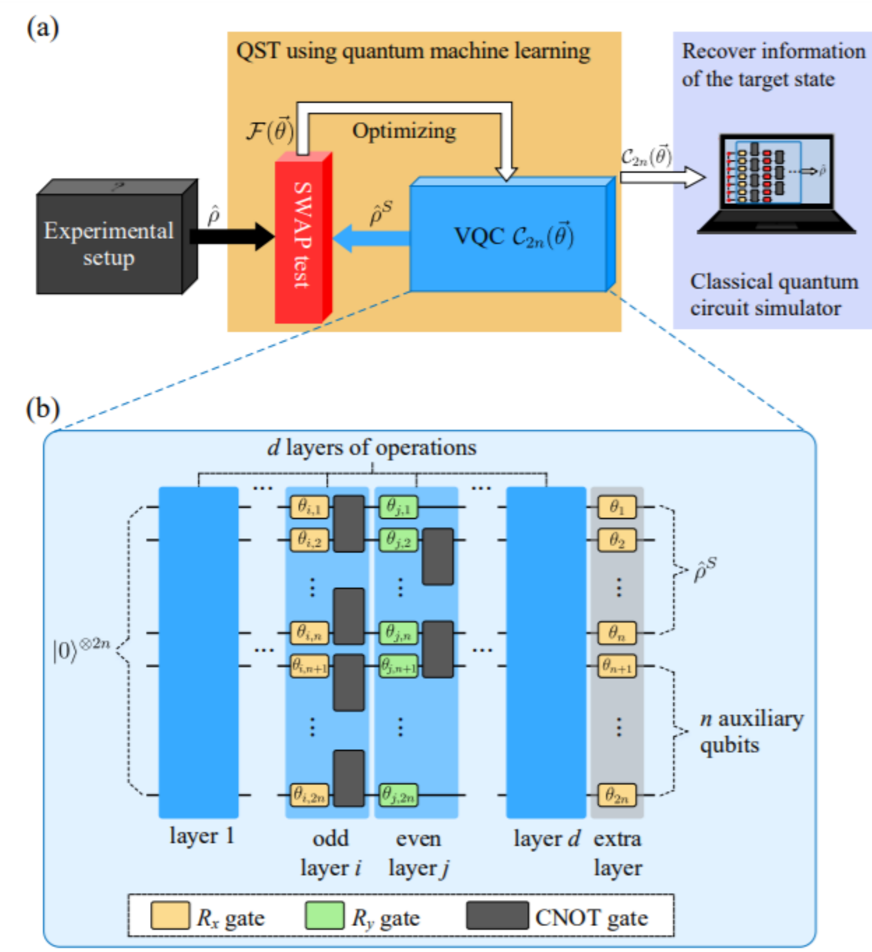
N= # features tau=#bits in binary rep





Quantum Circuits

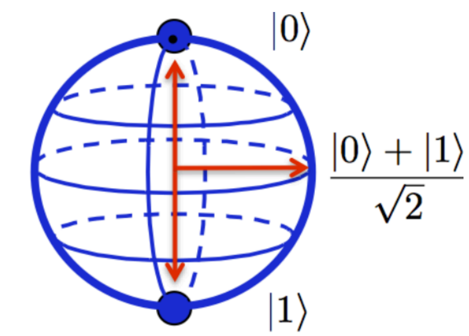




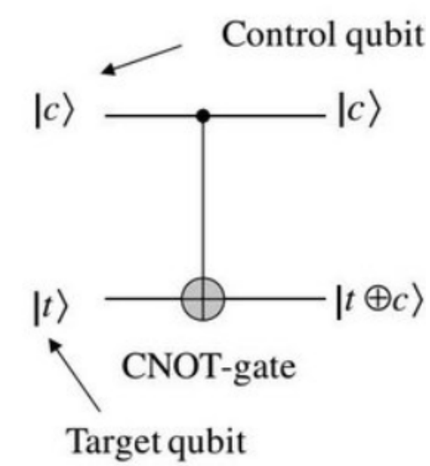
Need transition form classical to quantum:

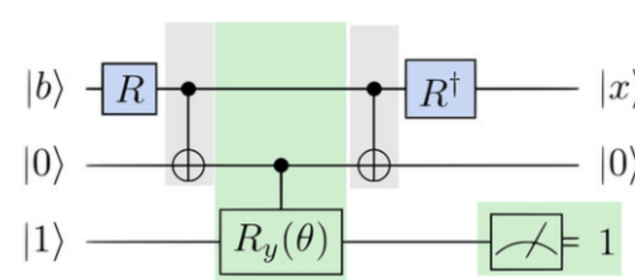
Classical Bit

<u>Classical</u>		<u>Quantum</u>
bits	⟷	qubits
gates	⟷	quantum gates
algorithms	⟷	quantum algorithms



Qubit





Single-Qubit Quantum Gates

Illustrative to write single-qubit operation as matrices

X-Gate: Quantum equivalent to classical NOT gate

$$|0\rangle \mapsto |1\rangle$$
$$|1\rangle \mapsto |0\rangle$$

⟶ Flips |0> to |1> and vice versa (hopping)

Represented by matrix $\quad \mathbf{X} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$

concretely $\quad \mathbf{X}|0\rangle = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} = |1\rangle$

It is unitary $\quad \mathbf{X}\mathbf{X}^\dagger = \mathbf{X}\mathbf{X}^{-1} = \mathbb{1}$

Z-Gate: Represented by matrix $\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$

Action

$$|0\rangle \mapsto \quad |0\rangle$$
$$|1\rangle \mapsto \text{-}\,|1\rangle$$

⟶ Eigenvalues +- 1

Note, the X, Y and Z gates are represented by the Pauli matrices

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \quad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad [\sigma_i, \sigma_j] = 2i\epsilon_{ijk}\sigma_k$$

$$\det \sigma_j \;=\; -1$$
$$\operatorname{tr} \sigma_j \;=\; 0.$$

$$\sigma_1^2 = \sigma_2^2 = \sigma_3^2 = -i\,\sigma_1\sigma_2\sigma_3 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = I$$

Hadamard gate: Matrix representation $\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$

Action:

$$|0\rangle \mapsto \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \longleftrightarrow |+\rangle := \frac{|0\rangle + |1\rangle}{\sqrt{2}}$$

$$|1\rangle \mapsto \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \longleftrightarrow |-\rangle := \frac{|0\rangle - |1\rangle}{\sqrt{2}}$$

Phase gate: Matrix representation $P_\phi := \begin{pmatrix} 1 & 0 \\ 0 & e^{i\phi} \end{pmatrix}$

With special phase values

$$S := P_{\pi/2} \quad T := P_{\pi/4} \quad R := P_{-\pi/4}$$

Summary of fixed 1-qubit gates:

Gate	Circuit representation	Matrix representation	Dirac representation
X	$-\boxed{X}-$	$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$	$\|1\rangle\langle 0\| + \|0\rangle\langle 1\|$
Y	$-\boxed{Y}-$	$\begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$	$i\|1\rangle\langle 0\| - i\|0\rangle\langle 1\|$
Z	$-\boxed{Z}-$	$\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$	$\|1\rangle\langle 0\| - \|0\rangle\langle 1\|$
H	$-\boxed{H}-$	$\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$	$\frac{1}{\sqrt{2}}(\|0\rangle + \|1\rangle)\langle 0\| + \frac{1}{\sqrt{2}}(\|0\rangle - \|1\rangle)\langle 1\|$
S	$-\boxed{S}-$	$\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$	$\frac{1}{\sqrt{2}}\|0\rangle\langle 0\| + \frac{1}{\sqrt{2}}i\|1\rangle\langle 1\|$
T	$-\boxed{T}-$	$\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 0 \\ 0 & e^{(-i\pi/4)} \end{pmatrix}$	$\frac{1}{\sqrt{2}}\|0\rangle\langle 0\| + \frac{1}{\sqrt{2}}e^{(-i\pi/4)}\|1\rangle\langle 1\|$

Quantum gate can be parametrised

Pauli rotations:

$$R_x(\theta) = e^{-i\frac{\theta}{2}\sigma_x} = \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & -i\sin\left(\frac{\theta}{2}\right) \\ -i\sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{pmatrix} = \cos\frac{\theta}{2}I - i\sin\frac{\theta}{2}X$$

$$R_y(\theta) = e^{-i\frac{\theta}{2}\sigma_y} = \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & -\sin\left(\frac{\theta}{2}\right) \\ \sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{pmatrix} = \cos\frac{\theta}{2}I - i\sin\frac{\theta}{2}Y$$

$$R_z(\theta) = e^{-i\frac{\theta}{2}\sigma_z} = \begin{pmatrix} e^{-i\frac{\theta}{2}} & 0 \\ 0 & e^{i\frac{\theta}{2}} \end{pmatrix} = \cos\frac{\theta}{2}I - i\sin\frac{\theta}{2}Z$$

generalised form via $R(\theta_1, \theta_2, \theta_3) = R_z(\theta_1)R_y(\theta_2)R_z(\theta_3)$

$$R(\theta_1, \theta_2, \theta_3) = \begin{pmatrix} e^{i(-\frac{\theta_1}{2}-\frac{\theta_3}{2})}\cos(\frac{\theta_2}{2}) & -e^{i(-\frac{\theta_1}{2}+\frac{\theta_3}{2})}\sin(\frac{\theta_2}{2}) \\ e^{i(\frac{\theta_1}{2}-\frac{\theta_3}{2})}\sin(\frac{\theta_2}{2}) & e^{i(\frac{\theta_1}{2}+\frac{\theta_3}{2})}\cos(\frac{\theta_2}{2}) \end{pmatrix}$$

Measurement process

Measurement process of a generic (normalised) qubit state $|\psi\rangle = \alpha_0|0\rangle + \alpha_1|1\rangle$

represented by projection onto eigenstates $P_0 = |0\rangle\langle 0|$ and $P_1 = |1\rangle\langle 1|$

Prob of measurement outcome 0 is then $p(0) = \mathrm{tr}(P_0|\psi\rangle\langle\psi|) = \langle\psi|P_0|\psi\rangle = |\alpha_0|^2$

and $p(1) = |\alpha_1|^2$

After measurement qubit is in state $|\psi\rangle \leftarrow \dfrac{P_0|\psi\rangle}{\sqrt{\langle\psi|P_0|\psi\rangle}} = |0\rangle$

The observable corresponding to a computational basis measurement is Pauli-Z observable

$$\sigma_z = |0\rangle\langle 0| - |1\rangle\langle 1|$$

(we know eigenvalues +1 for |0> and -1 for |1>)

The expectation value $\langle \sigma_z \rangle$ in a value in [-1, 1]. Its error can be estimated as sampling from a Bernoulli distribution.

Wald interval gives

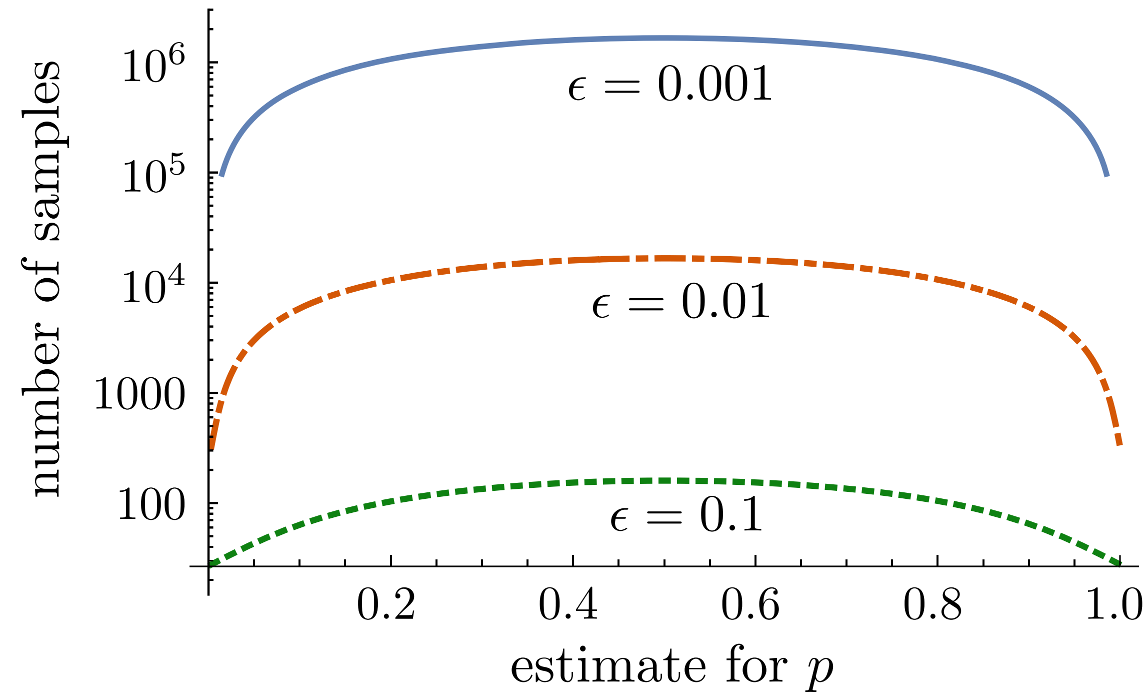
(suited for large s and p~0.5)

$$\epsilon = z\sqrt{\frac{\hat{p}(1-\hat{p})}{S}}$$

share of sample in state 1

shots

stat. z-value



→ For $\epsilon = 0.1$ and conf level 99% one needs 167 samples

For $\epsilon = 0.01$ and conf level 99% one needs $17{,}000$ samples

→ Overall might need a large number of shots on quantum computer

This needs to be taken into account when comparing quantum and classical computers in terms of speedups and quantum advantage

The Bloch Sphere

Since $|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$ with $|\alpha|^2 + |\beta|^2 = 1$

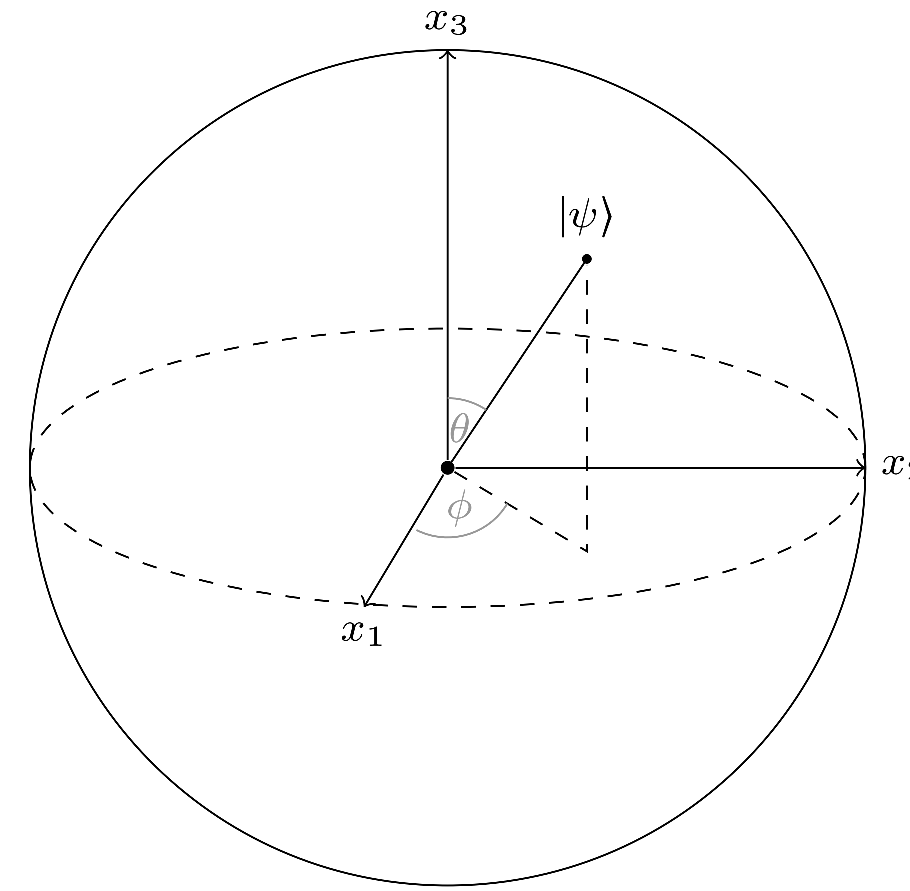
one can find angles such that

$$\alpha = e^{i\gamma}\cos\frac{\theta}{2} \qquad \beta = e^{i\delta}\sin\frac{\theta}{2}$$

Thus, with $\phi = \delta - \gamma$ single qubit can be parametrised as

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

where a global imaginary phase has no measurable effect and can be omitted.



$(\sin\theta\cos\phi, \sin\theta\sin\phi, \cos\phi)$

2-qubit states

Are built by tensor products, each qubit can be in state |0> or in state |1>

So, for two qubits we have four possibilities:

$$|0\rangle \otimes |0\rangle\,, |0\rangle \otimes |1\rangle\,, |1\rangle \otimes |0\rangle\,, |1\rangle \otimes |1\rangle$$

that we denote

$$|0\rangle\,|0\rangle\,, |0\rangle\,|1\rangle\,, |1\rangle\,|0\rangle\,, |1\rangle\,|1\rangle$$

or

$$|00\rangle\,, |01\rangle\,, |10\rangle\,, |11\rangle$$

We can have superposition as a generic state

$$|\psi\rangle = \alpha_{00}\,|00\rangle + \alpha_{01}\,|01\rangle + \alpha_{10}\,|10\rangle + \alpha_{11}\,|11\rangle$$

with complex coefficients such that $\sum_{x,y=0}^{1} |\alpha_{xy}|^2 = 1$

2-qubit states

Furthermore, we can express the state as a vector

$$\begin{pmatrix} \alpha_{00} \\ \alpha_{01} \\ \alpha_{10} \\ \alpha_{11} \end{pmatrix}$$

For which we find the inner products

$$\langle 00|00\rangle = \langle 01|01\rangle = \langle 10|10\rangle = \langle 11|11\rangle = 1$$

$$\langle 00|01\rangle = \langle 00|10\rangle = \langle 00|11\rangle = \cdots = \langle 11|00\rangle = 0$$

A 2-qubit quantum gate is a unitary matrix U of size 4 x 4

2-qubit gates

CNOT gate:

unitary matrix representation

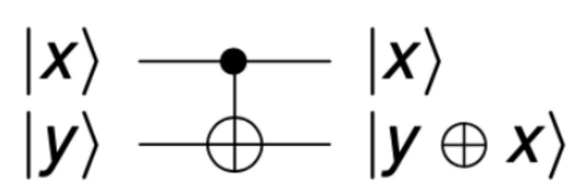
$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

In words: if the first qubit is |0> nothing changes. If it is |1> we flip the second bit (and first stays the same)

Action: $|00\rangle \to |00\rangle \quad |01\rangle \to |01\rangle$

$|10\rangle \to |11\rangle \quad |11\rangle \to |10\rangle$

As a gate: $x, y \in \{0, 1\} \longrightarrow$



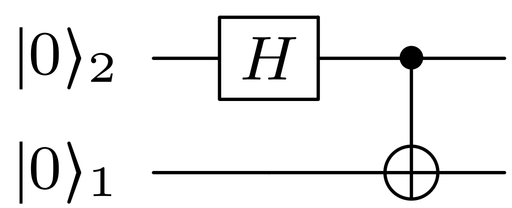
- A set of gates that can approximate any quantum operation
  -> Universal quantum computer

e.g. Rotation gates $R_x(\theta), R_y(\theta), R_z(\theta)$ + phase shift gate $P(\varphi)$ + CNOT

The CNOT gate is an extremely important gate

- It realises conditional probabilities

- It creates entanglement



read circuit from left to right →

$$\mathrm{CNOT}((H_2 \otimes \mathbb{1}_1)(|0\rangle_2 \otimes |0\rangle_1))$$

$$\mathrm{CNOT}(\frac{1}{\sqrt{2}}|0\rangle_2 \otimes |0\rangle_1 + \frac{1}{\sqrt{2}}|1\rangle_2 \otimes |0\rangle_1) = \frac{1}{\sqrt{2}}(|0\rangle_2 \otimes |0\rangle_1 + |1\rangle_2 \otimes |1\rangle_1)$$

Bell state (fully entangled)

- It can copy classical information, because

$$|00\rangle \rightarrow |00\rangle$$

$$|10\rangle \rightarrow |11\rangle$$

- Constructs other control gates

SWAP gate

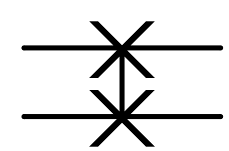
Can swap two qubits.

In basis $|00\rangle, |01\rangle, |10\rangle, |11\rangle$

it is represented by

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

In gate notation:



Can be decomposed by Pauli operators

$$\mathrm{SWAP} = \frac{I \otimes I + X \otimes X + Y \otimes Y + Z \otimes Z}{2}$$

N-qubit states

When we have n qubits, each of them can be in state |0> or |1>

Thus for n qubit states we have 2^n possibilities:

$$|00\ldots0\rangle, |00\ldots1\rangle, \ldots, |11\ldots1\rangle$$

or simply

$$|0\rangle, |1\rangle, \ldots, |2^n-1\rangle$$

A generic state of the system will be

$$|\psi\rangle = \alpha_0|0\rangle + \alpha_1|1\rangle + \ldots + \alpha_{2^n-1}|2^n-1\rangle$$

With complex coefficients, such that $\sum_{i=0}^{2^n-1} |\alpha_i|^2 = 1$

Suppose we have the N qubit state

$$|\psi\rangle = \alpha_0 |0\rangle + \alpha_1 |1\rangle + \ldots + \alpha_{2^n-1} \left|2^n - 1\right\rangle$$

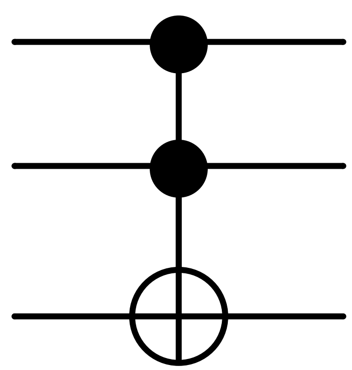
If we measure all its qubits, we obtain:

- 0 with probability $|\alpha_0|^2$ and the new state will be $|0\ldots00\rangle$
- 1 with probability $|\alpha_1|^2$ and the new state will be $|0\ldots01\rangle$
- ...
- $2^n - 1$ with probability $|\alpha_{2^n-1}|^2$ and the new state is $|1\ldots11\rangle$

Completely analogous to 1 and 2 qubit situation but now with 2^n possibilities

Toffoli gate (CCNOT)

controls from two qubits



Matrix representation

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

Truth table

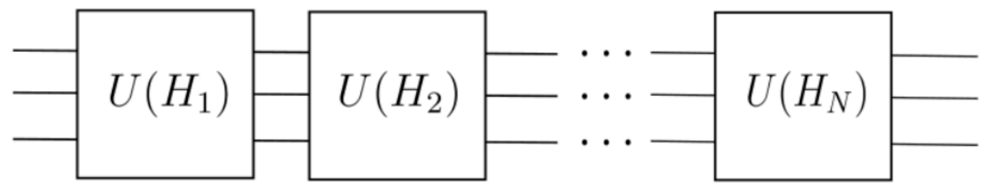
INPUT			OUTPUT		
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	0	1
1	1	0	1	1	1
1	1	1	1	1	0

Toffoli gate can also be decomposed into Pauli operators

$$\mathrm{Toff} = e^{i\frac{\pi}{8}(I-Z_1)(I-Z_2)(I-X_3)} = e^{-i\frac{\pi}{8}(I-Z_1)(I-Z_2)(I-X_3)}$$

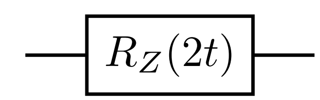
Example: Turning a Hamiltonian term into a gate

Recall $H = H_1 + H_2 + \cdots + H_N$



Assume, universal gate operations on device are $\{H, R_Z, CX\}$

Example 1 Assume $H_1 = Z \longrightarrow U = e^{-iZt} \longrightarrow$

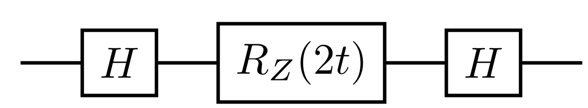


$R_Z(\theta) = e^{-i\frac{\theta}{2}Z}$

Example 2 Assume $H_2 = X \longrightarrow$ Since $HXH = Z \Rightarrow X = HZH$

$\longrightarrow U = He^{-iZt}H$ (proof via CBH Formula)

$\longrightarrow$



Example 3 $H = Z \otimes Z$ note $e^{-Z\otimes Zt} \neq e^{-iZt} \otimes e^{-iZt}$

with $(Z \otimes Z)^2 = \mathbb{I}$ one finds $e^{i(Z\otimes Z)t} = \cos(t)\mathbb{I} - i\sin(t) Z \otimes Z$

for the action on states we find

$$e^{i(Z\otimes Z)t} |00\rangle = (\cos(t)\mathbb{I} - i\sin(t)Z \otimes Z) |00\rangle = (\cos(t) - i\sin(t)) |00\rangle$$

$$e^{i(Z\otimes Z)t} |11\rangle = (\cos(t)\mathbb{I} - i\sin(t)Z \otimes Z) |11\rangle = (\cos(t) - i\sin(t)) |11\rangle$$

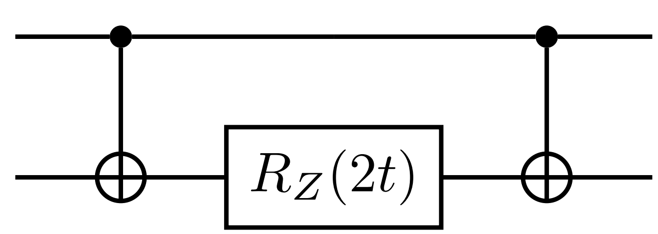
$$e^{i(Z\otimes Z)t} |01\rangle = \cos(t) |01\rangle - i\sin(t) Z|0\rangle \otimes Z|1\rangle = (\cos(t) + i\sin(t)) |01\rangle$$

which can be written in matrix form as

$$e^{i(Z\otimes Z)t} = \begin{bmatrix} e^{-it} & 0 & 0 & 0 \\ 0 & e^{it} & 0 & 0 \\ 0 & 0 & e^{it} & 0 \\ 0 & 0 & 0 & e^{-it} \end{bmatrix} \begin{matrix} |00\rangle \\ |01\rangle \\ |10\rangle \\ |11\rangle \end{matrix}$$

if # of 1 is even one gets -
if #of 1 is odd one gets + (parity of state)

circuit that implements that



with $R_Z(2t) = \begin{bmatrix} e^{-it} & 0 \\ 0 & e^{it} \end{bmatrix}$

Overlap of Quantum States

SWAP test:

Is a way to extract $|\langle a|b\rangle|^2$ of tensor product state $|a\rangle \otimes |b\rangle = |a\rangle|b\rangle$

One adds an ancilla qubit $|0\rangle|a\rangle|b\rangle$

then apply an H to the ancilla $\longrightarrow$ $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)|a\rangle|b\rangle$

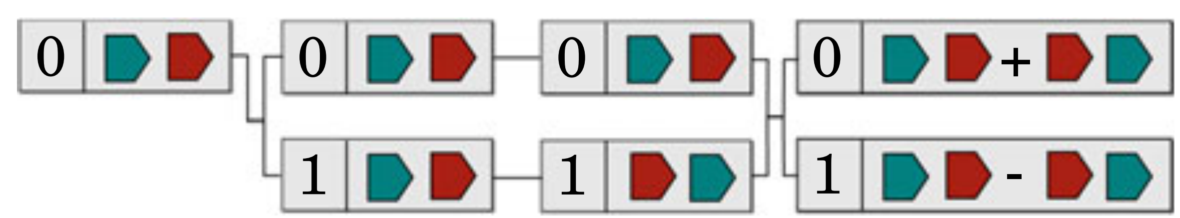
apply SWAP gate to |a> and |b> condition to ancilla being in state 1 $\longrightarrow$ $\frac{1}{\sqrt{2}}(|0\rangle|a\rangle|b\rangle + |1\rangle|a\rangle|b\rangle)$ (swap)

another H on the ancilla $\longrightarrow$ $|\psi\rangle = \frac{1}{2}|0\rangle \otimes (|a\rangle|b\rangle + |b\rangle|a\rangle) + \frac{1}{2}|1\rangle \otimes (|a\rangle|b\rangle - |b\rangle|a\rangle)$

Measure ancilla. Probability it is in 0 is:

$$p_0 = \frac{1}{2} - \frac{1}{2}|\langle a|b\rangle|^2 \longrightarrow |\langle a|b\rangle|^2 = 1 - 2p_0$$

overlap between both states



Hadamard test:

Elegant way to measure overlap/scalar product of quantum states

Start with superposition of ancilla and 1 register $|\psi\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle|a\rangle + |1\rangle|b\rangle\right)$

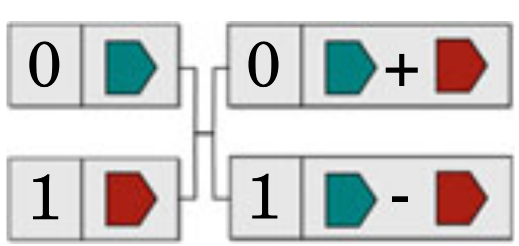
Then apply H on ancilla $|\psi\rangle = \frac{1}{2}|0\rangle \otimes (|a\rangle + |b\rangle) + \frac{1}{2}|1\rangle \otimes (|a\rangle - |b\rangle)$

The acceptance probability of ancilla to be in 0

$$\begin{aligned} p(0) &= \frac{1}{4}\left(\langle a| + \langle b|\right)\left(|a\rangle + |b\rangle\right), \\ &= \frac{1}{4}(2 + \langle a|b\rangle + \langle b|\mathbf{a}\rangle, \\ &= \frac{1}{2} + \frac{1}{2}\,\mathrm{Re}(\langle a|b\rangle). \end{aligned}$$

Starting with ancilla in $|-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - i|1\rangle)$ gives

$$\begin{aligned} p(0) &= \frac{1}{4}\left(\langle a| - i\langle b|\right)\left(|a\rangle + i|b\rangle\right), \\ &= \frac{1}{4}(2 - i\langle b|a\rangle + i\langle a|b\rangle, \\ &= \frac{1}{2} - \frac{1}{2}\,\mathrm{Im}(\langle a|b\rangle). \end{aligned}$$



Grover Algorithm

- Well-known algorithm to give quadratic speedup in finding element in unordered list. Classically, this takes on average K/2 steps in a list of length K…

- Idea is based on amplitude amplification. One encodes the elements as basis states and iteratively increases the value of the amplitude of the element of interest.

- For example:

Basis state — Ancilla — Basis encoded entries

$$|\psi\rangle = \alpha_0|000\rangle|0\rangle|\mathrm{e}_0\rangle + \alpha_1|001\rangle|0\rangle|\mathrm{e}_1\rangle + \alpha_2|010\rangle|0\rangle|\mathrm{e}_2\rangle + \alpha_3|011\rangle|0\rangle|\mathrm{e}_3\rangle + \alpha_4|100\rangle|0\rangle|\mathrm{e}_4\rangle + \alpha_5|101\rangle|0\rangle|\mathrm{e}_5\rangle + \alpha_6|110\rangle|0\rangle|\mathrm{e}_6\rangle + \alpha_7|111\rangle|0\rangle|\mathrm{e}_7\rangle$$

Gover algorithm iteratively increases amplitdue of element of interest

requires $\sqrt{2^n}$ iterations for n qubits to get to $|\alpha_3|^2 \approx 1$

$$|\psi'\rangle = \alpha_0|000\rangle|0\rangle|\mathrm{e}_0\rangle + \alpha_1|001\rangle|0\rangle|\mathrm{e}_1\rangle + \alpha_2|010\rangle|0\rangle|\mathrm{e}_2\rangle + \alpha_3|011\rangle|\mathbf{1}\rangle|\mathrm{e}_3\rangle + \alpha_4|100\rangle|0\rangle|\mathrm{e}_4\rangle + \alpha_5|101\rangle|0\rangle|\mathrm{e}_5\rangle + \alpha_6|110\rangle|0\rangle|\mathrm{e}_6\rangle + \alpha_7|111\rangle|0\rangle|\mathrm{e}_7\rangle$$

element of interest