Quantum Computing Seminar 2

YongHyun "Aeren" An

Samsung Research

September 9, 2024
September 25, 2024

History

From **Wikipedia**.

History

From **Wikipedia**.

- Both computer science and quantum mechanics had practical usage in World War II;
  1. computers played a major role in wartime cryptography, and
  2. quantum physics was essential for the nuclear physics used in the Manhattan Project.

History

From **Wikipedia**.

- Both computer science and quantum mechanics had practical usage in World War II;
  1. computers played a major role in wartime cryptography, and
  2. quantum physics was essential for the nuclear physics used in the Manhattan Project.
- When digital computers became faster, physicists faced an exponential increase in overhead when simulating quantum dynamics.

History

From **Wikipedia**.

- Both computer science and quantum mechanics had practical usage in World War II;
  1. computers played a major role in wartime cryptography, and
  2. quantum physics was essential for the nuclear physics used in the Manhattan Project.
- When digital computers became faster, physicists faced an exponential increase in overhead when simulating quantum dynamics.
- It prompted Yuri Marnin and Richard Feynman to independently suggest that hardware based on quantum phenomena might be more efficient for computer simulation.

Quantum Computer

- **(Classical) computer** stores information with a sequence of **(classical) bits**.

Quantum Computer

- **(Classical) computer** stores information with a sequence of **(classical) bits**.
- **Quantum computer** stores information with a sequence of **quantum bits** or **qubits**.

Quantum Computer

- **(Classical) computer** stores information with a sequence of **(classical) bits**.
- **Quantum computer** stores information with a sequence of **quantum bits** or **qubits**.
- It has been proven that the **(classical) Turing machine** and **quantum Turing machine** are equivalent. The power of quantum computer lies in complexity, not in decidability.

Quantum Computer

- **(Classical) computer** stores information with a sequence of **(classical) bits**.
- **Quantum computer** stores information with a sequence of **quantum bits** or **qubits**.
- It has been proven that the **(classical) Turing machine** and **quantum Turing machine** are equivalent. The power of quantum computer lies in complexity, not in decidability.
- We need to store $\Theta(2^N)$ complex numbers in order to simulate a quantum computer with N qubits. It is very inefficient to simulate quantum computer with a (classical) computer, but it can be done.

Qubit

Qubit

- A bit stores a boolean value of 0 or 1. We can directly "look" at the value a bit stores.

Qubit

- A bit stores a boolean value of 0 or 1. We can directly "look" at the value a bit stores.
- A qubit stores a 2-dimensional complex vector $v = (a, b)$ with $|v| = \sqrt{|a|^2 + |b|^2} = 1$. We **cannot** look at these values.

Qubit

- A bit stores a boolean value of 0 or 1. We can directly "look" at the value a bit stores.
- A qubit stores a 2-dimensional complex vector $v = (a, b)$ with $|v| = \sqrt{|a|^2 + |b|^2} = 1$. We **cannot** look at these values.
- Instead, we can **measure** a qubit. It will collapse into the value of 0 with probability $|a|^2$, and into the value of 1 with probability $|b|^2$.

Qubit

- A bit stores a boolean value of 0 or 1. We can directly "look" at the value a bit stores.
- A qubit stores a 2-dimensional complex vector $v = (a, b)$ with $|v| = \sqrt{|a|^2 + |b|^2} = 1$. We **cannot** look at these values.
- Instead, we can **measure** a qubit. It will collapse into the value of 0 with probability $|a|^2$, and into the value of 1 with probability $|b|^2$.

Reminder: notation for Qubit

$$(a, b) = \begin{bmatrix} a \\ b \end{bmatrix} = a \cdot |0\rangle + b \cdot |1\rangle$$

Qubit

Example

Which of the following can be the state of a qubit?

1. $a = \left(\frac{1}{3} + \frac{2}{3}\mathrm{i}, -\frac{2}{3}\right)$
2. $b = \left(1, 3 \cdot e^{2\pi \mathrm{i}/3}\right)$
3. $c = \left(1/\sqrt{2}, 1/\sqrt{2}\right)$
4. $d = \left(1/\sqrt{2}, -1/\sqrt{2}\right)$

Qubit

Example

Which of the following can be the state of a qubit?

1. $a = \left(\frac{1}{3} + \frac{2}{3}\mathrm{i}, -\frac{2}{3}\right)$
2. $b = \left(1, 3 \cdot e^{2\pi\mathrm{i}/3}\right)$
3. $c = \left(1/\sqrt{2}, 1/\sqrt{2}\right)$
4. $d = \left(1/\sqrt{2}, -1/\sqrt{2}\right)$

1. $|a| = \sqrt{\left|\frac{1}{3} + \frac{2}{3}\mathrm{i}\right|^2 + \left|-\frac{2}{3}\right|^2} = \sqrt{\left|\frac{1}{3}\right|^2 + \left|\frac{2}{3}\right|^2 + \left|-\frac{2}{3}\right|^2} = 1$

Qubit

Example

Which of the following can be the state of a qubit?

1. $a = \left(\frac{1}{3} + \frac{2}{3}\mathrm{i}, -\frac{2}{3}\right)$
2. $b = \left(1, 3 \cdot e^{2\pi\mathrm{i}/3}\right)$
3. $c = \left(1/\sqrt{2}, 1/\sqrt{2}\right)$
4. $d = \left(1/\sqrt{2}, -1/\sqrt{2}\right)$

1. $|a| = \sqrt{\left|\frac{1}{3} + \frac{2}{3}\mathrm{i}\right|^2 + \left|-\frac{2}{3}\right|^2} = \sqrt{\left|\frac{1}{3}\right|^2 + \left|\frac{2}{3}\right|^2 + \left|-\frac{2}{3}\right|^2} = 1$
2. $|b| = \sqrt{|1|^2 + |3|^2} = \sqrt{10} \neq 1$

Qubit

Example

Which of the following can be the state of a qubit?

1. $a = \left(\frac{1}{3} + \frac{2}{3}\mathrm{i}, -\frac{2}{3}\right)$
2. $b = \left(1, 3 \cdot e^{2\pi \mathrm{i}/3}\right)$
3. $c = \left(1/\sqrt{2}, 1/\sqrt{2}\right)$
4. $d = \left(1/\sqrt{2}, -1/\sqrt{2}\right)$

1. $|a| = \sqrt{\left|\frac{1}{3} + \frac{2}{3}\mathrm{i}\right|^2 + \left|-\frac{2}{3}\right|^2} = \sqrt{\left|\frac{1}{3}\right|^2 + \left|\frac{2}{3}\right|^2 + \left|-\frac{2}{3}\right|^2} = 1$
2. $|b| = \sqrt{|1|^2 + |3|^2} = \sqrt{10} \neq 1$
3. $|c| = \sqrt{\left|1/\sqrt{2}\right|^2 + \left|1/\sqrt{2}\right|^2} = 1$

Qubit

Example

Which of the following can be the state of a qubit?

1. $a = \left(\frac{1}{3} + \frac{2}{3}\mathrm{i}, -\frac{2}{3}\right)$
2. $b = \left(1, 3 \cdot e^{2\pi\mathrm{i}/3}\right)$
3. $c = \left(1/\sqrt{2}, 1/\sqrt{2}\right)$
4. $d = \left(1/\sqrt{2}, -1/\sqrt{2}\right)$

1. $|a| = \sqrt{\left|\frac{1}{3} + \frac{2}{3}\mathrm{i}\right|^2 + \left|-\frac{2}{3}\right|^2} = \sqrt{\left|\frac{1}{3}\right|^2 + \left|\frac{2}{3}\right|^2 + \left|-\frac{2}{3}\right|^2} = 1$
2. $|b| = \sqrt{|1|^2 + |3|^2} = \sqrt{10} \neq 1$
3. $|c| = \sqrt{\left|1/\sqrt{2}\right|^2 + \left|1/\sqrt{2}\right|^2} = 1$
4. $|d| = \sqrt{\left|1/\sqrt{2}\right|^2 + \left|-1/\sqrt{2}\right|^2} = 1$

Qubit

Definition(Measurement)

Given a qubit with state v, **measuring** it yields

- the value 0 with probability $\mathcal{P}_0\left(v\right) := \left|\langle 0| \cdot v\right|^2$, and
- the value 1 with probability $\mathcal{P}_1\left(v\right) := \left|\langle 1| \cdot v\right|^2$.

Qubit

Definition(Measurement)

Given a qubit with state v, **measuring** it yields

- the value 0 with probability $\mathcal{P}_0\left(v\right) := \left|\langle 0| \cdot v\right|^2$, and
- the value 1 with probability $\mathcal{P}_1\left(v\right) := \left|\langle 1| \cdot v\right|^2$.

Measuring $|0\rangle$ and $|1\rangle$

- $\mathcal{P}_0\left(|0\rangle\right) = \left|\langle 0|0\rangle\right|^2 = 1$
- $\mathcal{P}_1\left(|0\rangle\right) = \left|\langle 1|0\rangle\right|^2 = 0$
- $\mathcal{P}_0\left(|1\rangle\right) = \left|\langle 0|1\rangle\right|^2 = 0$
- $\mathcal{P}_1\left(|1\rangle\right) = \left|\langle 1|1\rangle\right|^2 = 1$

Qubit

Definition

- $|+\rangle := \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$
- $|-\rangle := \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle$

Qubit

Definition

- $|+\rangle := \frac{1}{\sqrt{2}} |0\rangle + \frac{1}{\sqrt{2}} |1\rangle$
- $|-\rangle := \frac{1}{\sqrt{2}} |0\rangle - \frac{1}{\sqrt{2}} |1\rangle$

Measuring $|+\rangle$ and $|-\rangle$

- $\mathcal{P}_0 (|+\rangle) = |\langle 0|+\rangle|^2 = \frac{1}{2}$
- $\mathcal{P}_1 (|+\rangle) = |\langle 1|+\rangle|^2 = \frac{1}{2}$
- $\mathcal{P}_0 (|-\rangle) = |\langle 0|-\rangle|^2 = \frac{1}{2}$
- $\mathcal{P}_1 (|-\rangle) = |\langle 1|-\rangle|^2 = \frac{1}{2}$

$|+\rangle$ and $|-\rangle$ are indistinguishable by measurement!

Qiskit Examples: Qubit

```python
from qiskit.quantum_info import Statevector
from numpy import sqrt, exp, pi

ket_a     = Statevector( [ 1/3 + 2j/3, -2/3              ] )
ket_b     = Statevector( [ 1          ,  3*exp(3j * pi) ] )
ket_plus  = Statevector( [ 1/sqrt(2) ,  1/sqrt(2)       ] )
ket_minus = Statevector( [ 1/sqrt(2) , -1/sqrt(2)       ] )
```

```python
ket_a, ket_b, ket_plus, ket_minus
```

```
(Statevector([ 0.33333333+0.66666667j, -0.66666667+0.j        ],
             dims=(2,)),
 Statevector([ 1.+0.00000000e+00j, -3.+1.10218212e-15j],
             dims=(2,)),
 Statevector([0.70710678+0.j, 0.70710678+0.j],
             dims=(2,)),
 Statevector([ 0.70710678+0.j, -0.70710678+0.j],
             dims=(2,)))
```

```python
ket_a.is_valid(), ket_b.is_valid(), ket_plus.is_valid(), ket_minus.is_valid()
```

```
(True, False, True, True)
```

```python
ket_a.draw("latex")
```

$$(\frac{1}{3} + \frac{2i}{3})|0\rangle - \frac{2}{3}|1\rangle$$

```python
ket_plus.draw("latex")
```

$$\frac{\sqrt{2}}{2}|0\rangle + \frac{\sqrt{2}}{2}|1\rangle$$

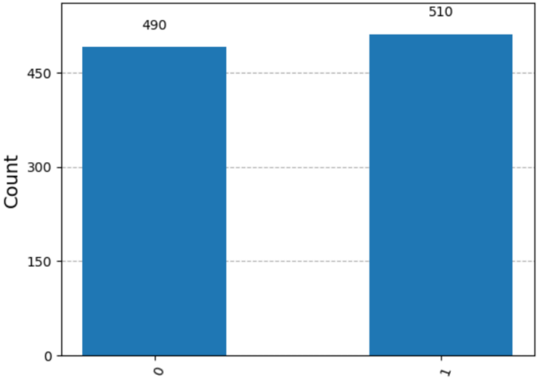
```python
ket_minus.draw("latex")
```

$$\frac{\sqrt{2}}{2}|0\rangle - \frac{\sqrt{2}}{2}|1\rangle$$

Qiskit Examples: Qubit

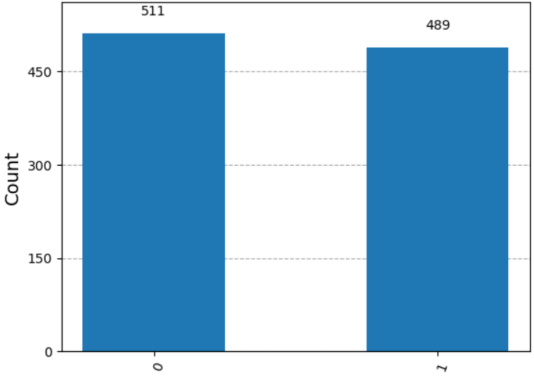
```
from qiskit.visualization import plot_histogram

statistics = ket_plus.sample_counts(1000)
print(statistics)
plot_histogram(statistics)
```

```
{'0': 490, '1': 510}
```




```
statistics = ket_minus.sample_counts(1000)
print(statistics)
plot_histogram(statistics)
```

```
{'0': 511, '1': 489}
```



Unitary Operation

Unitary Operation

- In classical computing, we were allowed to operate on bits via logic gates.

Unitary Operation

- In classical computing, we were allowed to operate on bits via logic gates.
- In quantum computing, the set of allowed operations is precisely the set of unitary operations.

Unitary Operation

- In classical computing, we were allowed to operate on bits via logic gates.
- In quantum computing, the set of allowed operations is precisely the set of unitary operations.

Reminder: Unitary Matrix

Let M be a unitary matrix.

- $M \cdot M^H = I$
- The set of rows/columns of M forms an orthonormal basis.
- M preserves inner product. i.e. $\langle M \cdot u, M \cdot v \rangle = \langle u, v \rangle$
- In particular, $|M \cdot u| = |u|$.

Unitary Operation

Pauli operations

$$X := \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, Y := \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, Z := \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

Unitary Operation

Pauli operations

$$X := \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, Y := \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, Z := \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

- $X \cdot |0\rangle = |1\rangle, X \cdot |1\rangle = |0\rangle$, X is also called a **bit flip** or a **NOT** operation.

Unitary Operation

Pauli operations

$$X := \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, Y := \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, Z := \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

- $X \cdot |0\rangle = |1\rangle, X \cdot |1\rangle = |0\rangle$, X is also called a **bit flip** or a **NOT** operation.
- $Z \cdot |0\rangle = |0\rangle, Z \cdot |1\rangle = -|1\rangle$, Z is also called a **phase flip**.

Unitary Operation

Pauli operations

$$X := \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, Y := \begin{bmatrix} 0 & -\mathrm{i} \\ \mathrm{i} & 0 \end{bmatrix}, Z := \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

- $X \cdot |0\rangle = |1\rangle, X \cdot |1\rangle = |0\rangle$, X is also called a **bit flip** or a **NOT** operation.
- $Z \cdot |0\rangle = |0\rangle, Z \cdot |1\rangle = -|1\rangle$, Z is also called a **phase flip**.
- $X^2 = Y^2 = Z^2 = I$

Unitary Operation

Hadamard Operation

$$H := \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{bmatrix}$$

Unitary Operation

Hadamard Operation

$$H := \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{bmatrix}$$

- $H \cdot |0\rangle = |+\rangle$
- $H \cdot |1\rangle = |-\rangle$
- $H \cdot |+\rangle = |0\rangle$
- $H \cdot |-\rangle = |1\rangle$

Unitary Operation

Hadamard Operation

$$H := \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{bmatrix}$$

- $H \cdot |0\rangle = |+\rangle$
- $H \cdot |1\rangle = |-\rangle$
- $H \cdot |+\rangle = |0\rangle$
- $H \cdot |-\rangle = |1\rangle$
- Let v be a state of a qubit which is either $|+\rangle$ or $|-\rangle$. We cannot determine which one it is by directly measuring v, but we can distinguish them by measuring $H \cdot v$.

Unitary Operation

Phase Operations

$$P_\theta := \begin{bmatrix} 1 & 0 \\ 0 & e^{\mathrm{i}\theta} \end{bmatrix}, S := P_{\pi/2} = \begin{bmatrix} 1 & 0 \\ 0 & \mathrm{i} \end{bmatrix}, T := P_{\pi/4} = \begin{bmatrix} 1 & 0 \\ 0 & 1/\sqrt{2} + \mathrm{i}/\sqrt{2} \end{bmatrix}$$

Unitary Operation

Phase Operations

$$P_\theta := \begin{bmatrix} 1 & 0 \\ 0 & e^{\mathrm{i}\theta} \end{bmatrix}, S := P_{\pi/2} = \begin{bmatrix} 1 & 0 \\ 0 & \mathrm{i} \end{bmatrix}, T := P_{\pi/4} = \begin{bmatrix} 1 & 0 \\ 0 & 1/\sqrt{2} + \mathrm{i}/\sqrt{2} \end{bmatrix}$$

- $P_\theta \cdot |0\rangle = |0\rangle, P_\theta \cdot |1\rangle = e^{\mathrm{i}\theta} \cdot |1\rangle$

Unitary Operation

Phase Operations

$$P_\theta := \begin{bmatrix} 1 & 0 \\ 0 & e^{\mathrm{i}\theta} \end{bmatrix}, S := P_{\pi/2} = \begin{bmatrix} 1 & 0 \\ 0 & \mathrm{i} \end{bmatrix}, T := P_{\pi/4} = \begin{bmatrix} 1 & 0 \\ 0 & 1/\sqrt{2} + \mathrm{i}/\sqrt{2} \end{bmatrix}$$

- $P_\theta \cdot |0\rangle = |0\rangle, P_\theta \cdot |1\rangle = e^{\mathrm{i}\theta} \cdot |1\rangle$
- $I = P_0, Z = P_\pi$

Unitary Operation

Phase Operations

$$P_\theta := \begin{bmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{bmatrix}, S := P_{\pi/2} = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}, T := P_{\pi/4} = \begin{bmatrix} 1 & 0 \\ 0 & 1/\sqrt{2} + i/\sqrt{2} \end{bmatrix}$$

- $P_\theta \cdot |0\rangle = |0\rangle, P_\theta \cdot |1\rangle = e^{i\theta} \cdot |1\rangle$
- $I = P_0, Z = P_\pi$
- $\mathcal{P}_x(v) = \mathcal{P}_x(P_\theta(v))$ where $x = 0$ or 1

Unitary Operation

R Operation

$$R := H \cdot S \cdot H = \begin{bmatrix} 1/2 + \mathfrak{i}/2 & 1/2 - \mathfrak{i}/2 \\ 1/2 - \mathfrak{i}/2 & 1/2 + \mathfrak{i}/2 \end{bmatrix}$$

Unitary Operation

R Operation

$$R := H \cdot S \cdot H = \begin{bmatrix} 1/2 + \mathrm{i}/2 & 1/2 - \mathrm{i}/2 \\ 1/2 - \mathrm{i}/2 & 1/2 + \mathrm{i}/2 \end{bmatrix}$$

- $R^2 = X$

Unitary Operation

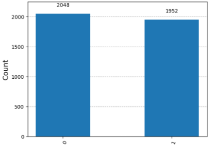
R Operation

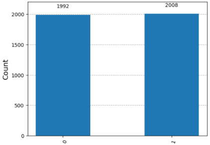
$$R := H \cdot S \cdot H = \begin{bmatrix} 1/2 + \mathfrak{i}/2 & 1/2 - \mathfrak{i}/2 \\ 1/2 - \mathfrak{i}/2 & 1/2 + \mathfrak{i}/2 \end{bmatrix}$$

- $R^2 = X$
- Note that this is not possible for classical operations.

Qiskit Examples: Unitary Operation

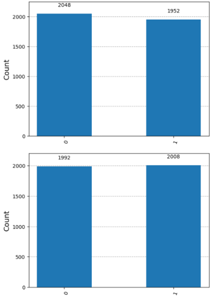
```
u = Statevector( [ 1/sqrt(2) ,  1/sqrt(2)      ] )
v = Statevector( [ 1/sqrt(2) , -1/sqrt(2)      ] )
display(plot_histogram(u.sample_counts(4000)))
display(plot_histogram(v.sample_counts(4000)))
```





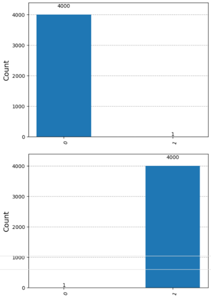
Qiskit Examples: Unitary Operation

```
u = Statevector( [ 1/sqrt(2) ,  1/sqrt(2)      ] )
v = Statevector( [ 1/sqrt(2) , -1/sqrt(2)      ] )
display(plot_histogram(u.sample_counts(4000)))
display(plot_histogram(v.sample_counts(4000)))
```



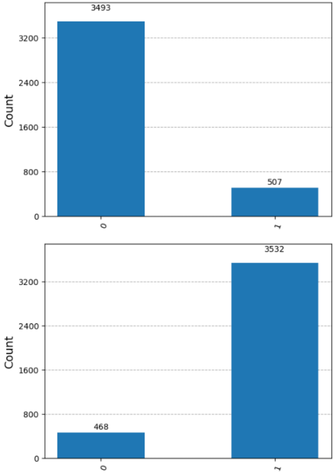

```
from qiskit.quantum_info import Operator

H = Operator([[1 / sqrt(2), 1 / sqrt(2)], [1 / sqrt(2), -1 / sqrt(2)]])

data_u = u.evolve(H).sample_counts(4000)
data_v = v.evolve(H).sample_counts(4000)
for c in "01":
    if c not in data_u:
        data_u.update({c : 1})
    if c not in data_v:
        data_v.update({c : 1})
display(plot_histogram(data_u))
display(plot_histogram(data_v))
```

Click to add a cell.



Qiskit Examples: Unitary Operation

```
S = Operator([[1, 0], [0, 1j]])
R = H.compose(S).compose(H)

u = Statevector([2/3 - 2j/3, 1j / 3])
display(plot_histogram(u.sample_counts(4000)))

u = u.evolve(R).evolve(R)

display(plot_histogram(u.sample_counts(4000)))
```



The End