SEENG 457/CSC 557

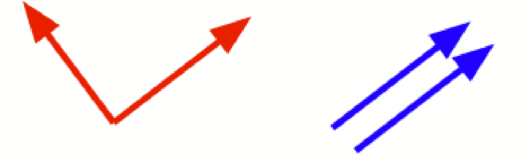
Lab 4: Qiskit and PennyLane (Quantum Teleportation)

Prashanti Priya Angara, Maziyar Khadivi
Contact email: mazy1996@uvic.ca

June 3, 2025

Bases

Basis States



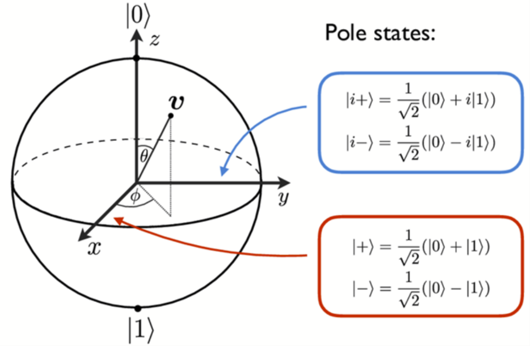
The general quantum state of a qubit can be represented by a linear superposition of its two orthonormal basis states $|x\rangle$ and $|y\rangle$ for example:

$$|\psi\rangle = \alpha|x\rangle + \beta|y\rangle$$

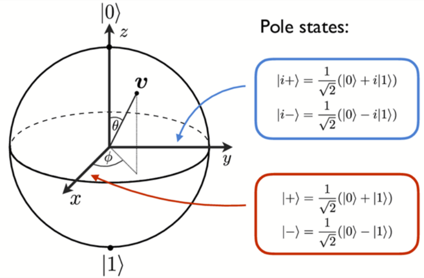
Computational Basis

- Computational Basis $\{|0\rangle, |1\rangle\}$ – for a single qubit system
- Measurement in the computational basis will only distinguish between the states $\{|0\rangle, |1\rangle\}$
- Sometimes measuring in another basis might be helpful
- Some other bases:
  - $\{|+\rangle, |-\rangle\}$
  - $\{|i+\rangle, |i-\rangle\}$

Some Common Bases



Pauli Measurements



Pauli Measurement	Unitary transformation
Z	$\mathbf{1}$
X	H
Y	$HS^\dagger$

Multi-Qubit Bases

- For a two qubit system, the computational basis is:

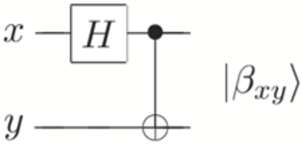
$$\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$$

- In general, for an n-qubit system, the computational basis is composed of 2^n elements:

$$\{|000...0\rangle_n, |000...1\rangle_n, ...|11...1\rangle_n\}$$

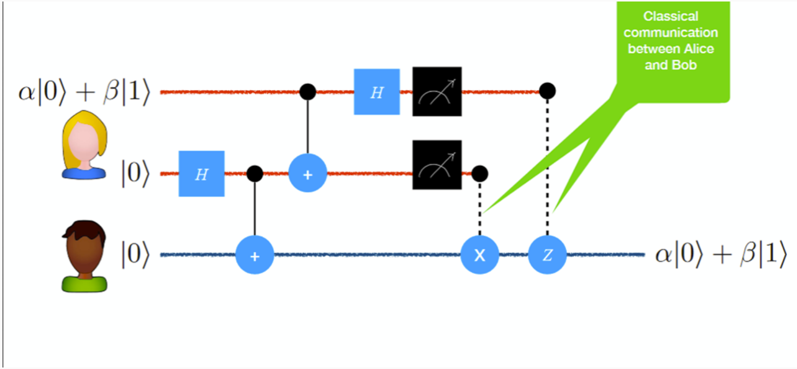
Bell Basis

In	Out
$\|00\rangle$	$(\|00\rangle + \|11\rangle)/\sqrt{2} \equiv \|\beta_{00}\rangle$
$\|01\rangle$	$(\|01\rangle + \|10\rangle)/\sqrt{2} \equiv \|\beta_{01}\rangle$
$\|10\rangle$	$(\|00\rangle - \|11\rangle)/\sqrt{2} \equiv \|\beta_{10}\rangle$
$\|11\rangle$	$(\|01\rangle - \|10\rangle)/\sqrt{2} \equiv \|\beta_{11}\rangle$

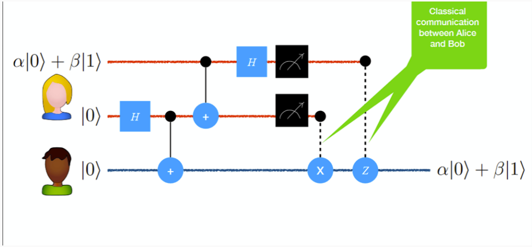


Teleportation: Another Way

Quantum Teleportation: Recap



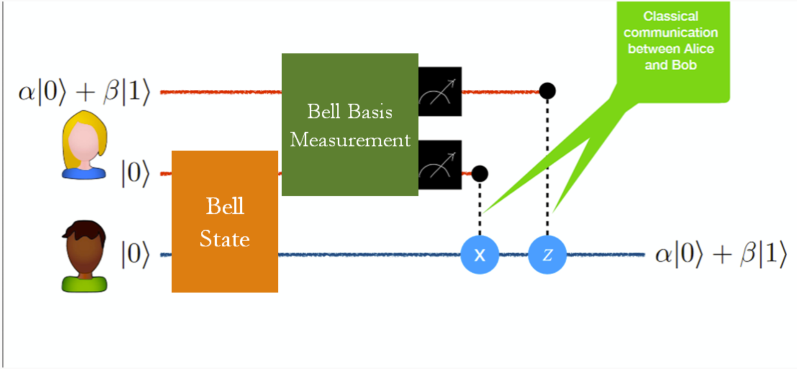
Quantum Teleportation: Recap



Final state before measurement:

$$\frac{1}{2}(|00\rangle \otimes \alpha|0\rangle + \beta|1\rangle) + \frac{1}{2}(|01\rangle \otimes \alpha|1\rangle + \beta|0\rangle) + \frac{1}{2}(|10\rangle \otimes \alpha|0\rangle - \beta|1\rangle) + \frac{1}{2}(|11\rangle \otimes \alpha|1\rangle - \beta|0\rangle)$$

Teleportation via Bell Basis

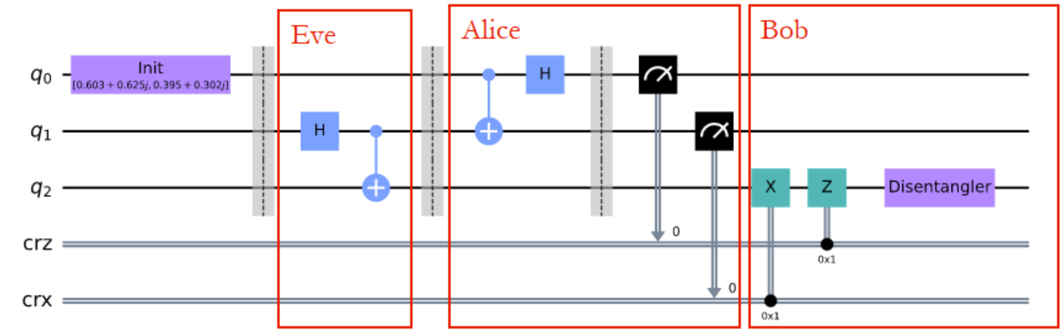


Principle of Deferred Measurement

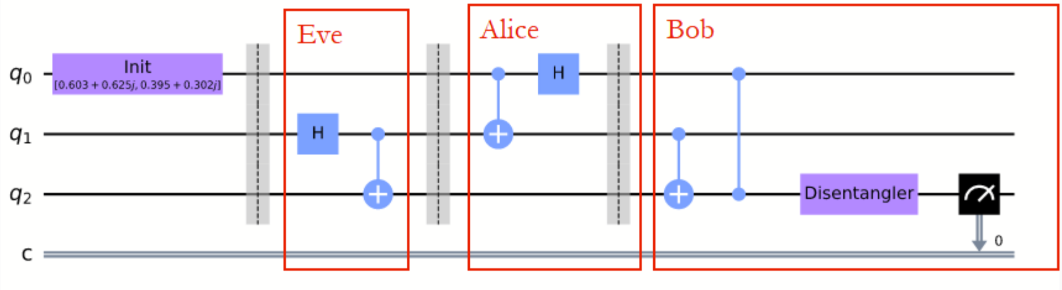
Deferred Measurement

The Deferred Measurement Principle is a result in quantum computing which states that delaying measurements until the end of a quantum computation doesn't affect the probability distribution of outcomes.

Teleportation Circuit



Modified Teleportation Circuit: Deferred Measurements



Teleportation in Qiskit

Mid-Circuit Measurements with Dynamic Circuits

- Incorporate real-time classical communication into quantum circuits
- Operations depend on data produced at run time
- Control flow with Qiskit:
  https://docs.quantum.ibm.com/build/classical-feedforward-and-control-flow

Teleportation in PennyLane

Mid-Circuit Measurements

PennyLane allows specifying measurements in the middle of the circuit. Quantum functions such as operations can then be conditioned on the measurement outcome of such mid-circuit measurements:

```
def my_quantum_function(x, y):
    qml.RY(x, wires=0)
    qml.CNOT(wires=[0, 1])
    m_0 = qml.measure(1)

    qml.cond(m_0, qml.RY)(y, wires=0)
    return qml.probs(wires=[0])
```

Deferred measurements in PennyLane

- A quantum function with mid-circuit measurements (defined using measure()) and conditional operations (defined using cond()) can be executed by applying the deferred measurement principle
- Performing true mid-circuit measurements and conditional operations is dependent on the quantum hardware and PennyLane device capabilities.

qml.cond[1]

- Quantum-compatible if-else conditionals: Condition quantum operations on parameters (results of mid-circuit measurements)
- When used with the `qjit()` decorator, this function allows for general if-elif-else constructs.

```
dev = qml.device("lightning.qubit", wires=1)

@qml.qjit
@qml.qnode(dev)
def circuit(x: float):
    def ansatz_true():
        qml.RX(x, wires=0)
        qml.Hadamard(wires=0)

    def ansatz_false():
        qml.RY(x, wires=0)

    qml.cond(x > 1.4, ansatz_true, ansatz_false)()

    return qml.expval(qml.PauliZ(0))
```

[1] https://docs.pennylane.ai/en/stable/code/api/pennylane.cond.html

References

Qiskit

- https://learning.quantum.ibm.com/course/basics-of-quantum-information/entanglement-in-action
- https://docs.quantum.ibm.com/build/classical-feedforward-and-control-flow
- https://www.ibm.com/quantum/blog/quantum-dynamic-circuits

PennyLane

- https://docs.pennylane.ai/en/stable/introduction/measurements.html
- https://docs.pennylane.ai/en/stable/code/api/pennylane.qjit.html
- https://docs.pennylane.ai/en/stable/code/api/pennylane.cond.html