CPEN 400Q / EECE 571Q Lecture 04
Your first quantum algorithms

Thursday 20 January 2022

Announcements

- Assignment 1 due 23:59 Thursday 27 Jan
  - Please put the word 'submission' or 'solution' in your branch name to help the autograder

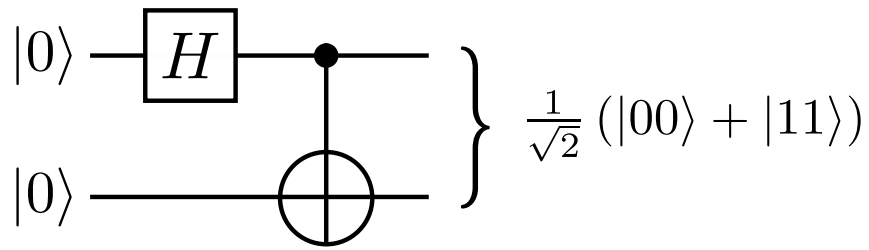
Last time

We expressed multi-qubit systems mathematically using the tensor product.

$$|01\rangle = |0\rangle \otimes |1\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ 0 \begin{pmatrix} 0 \\ 1 \end{pmatrix} \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

Last time

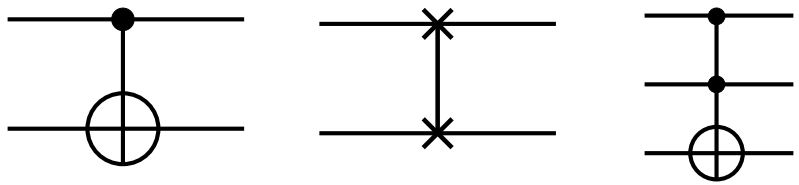
We learned what it meant for a multi-qubit state to be *entangled*.



$$\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \neq (\alpha|0\rangle + \beta|1\rangle) \otimes (\gamma|0\rangle + \delta|1\rangle)$$

Last time

We learned some two-qubit gates, (CNOT, SWAP, Toffoli)



We used them to write multi-qubit circuits in PennyLane:

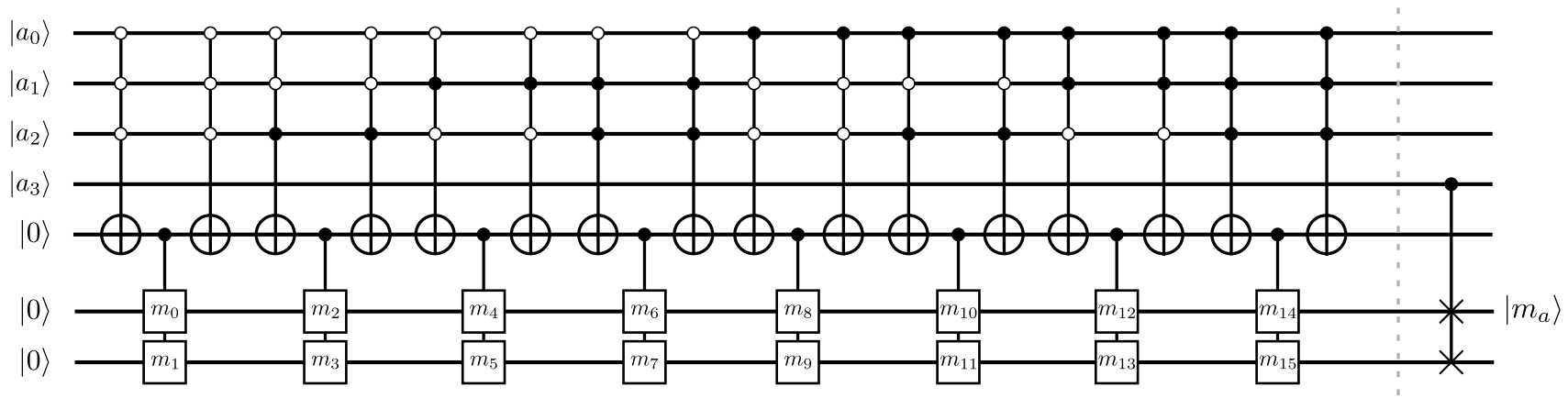
```
dev = qml.device('default.qubit', wires=3)

@qml.qnode(dev)
def qfunc(x, y):
    qml.Hadamard(wires=0)
    qml.RX(x, wires=1)
    qml.RY(y, wires=2)
    qml.Toffoli(wires=[0, 1, 2])
    return qml.expval(qml.PauliX(2))
```

Learning outcomes

- Make any gate a controlled gate
- Perform measurements on multiple qubits
- Implement the superdense coding protocol
- Teleport a qubit

Controlled unitary operations

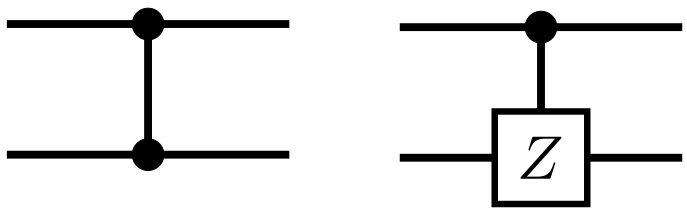
Any unitary operation can be turned into a controlled operation, controlled on any state.



Most common controls are controlled-on-$|1\rangle$ (filled circle), and controlled-on-$|0\rangle$ (empty circle).

Example: controlled-Z (CZ)

What does this operation do?



PennyLane: `qml.CZ`

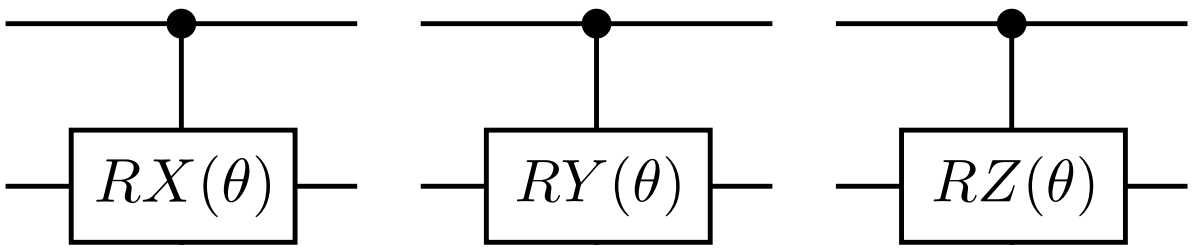
Image credit: Codebook node I.13

Example: controlled rotations (RX, RY, RZ)

Or this one?

$$CRY(\theta) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \cos\frac{\theta}{2} & -\sin\frac{\theta}{2} \\ 0 & 0 & \sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{pmatrix}$$

Circuit elements:



PennyLane: `qml.CRX`, `qml.CRY`, `qml.CRZ`

Image credit: Codebook node I.14

Controlled-U

There is a pattern here:

$$CNOT = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \quad CRY(\theta) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \cos\frac{\theta}{2} & -\sin\frac{\theta}{2} \\ 0 & 0 & \sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{pmatrix}$$

More generally,

$$CU = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & U_{00} & U_{01} \\ 0 & 0 & U_{10} & U_{11} \end{pmatrix} = \begin{pmatrix} I_2 & 0_2 \\ 0_2 & U \end{pmatrix}$$

... we don't want to be writing these matrices all the time.

Hands-on: qml.ctrl

Remember from last class, qml.adjoint:

```
@qml.qnode(dev)
def my_circuit():
    qml.S(wires=0)
    qml.adjoint(qml.S)(wires=0)
    return qml.sample()
```

There is a similar *transform* that allows us to perform arbitrary controlled operations (or entire quantum functions)!

```
@qml.qnode(dev)
def my_circuit():
    qml.S(wires=0)
    qml.ctrl(qml.S, control=1)(wires=0)
    return qml.sample()
```

Reversible circuits

A gate is reversible if we can invert (undo) it; we need to be able to get the outputs back from the inputs.

Not reversible: `XOR`, and `AND`.

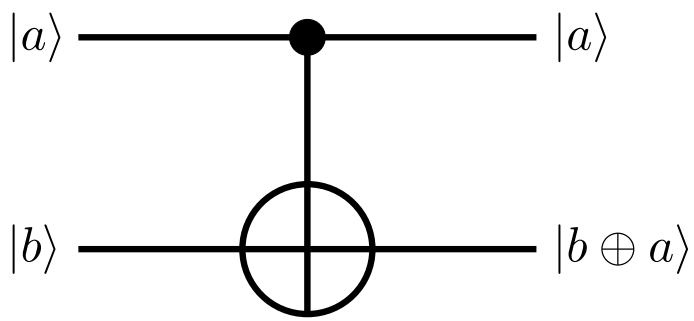
a	b	$a \oplus b$
0	0	0
0	1	1
1	0	1
1	1	0

a	b	ab
0	0	0
0	1	0
1	0	0
1	1	1

Quantum computations are unitary; by definition unitaries are reversible. To implement the above non-reversible operations, we need to construct reversible versions of them.

Reversible circuits

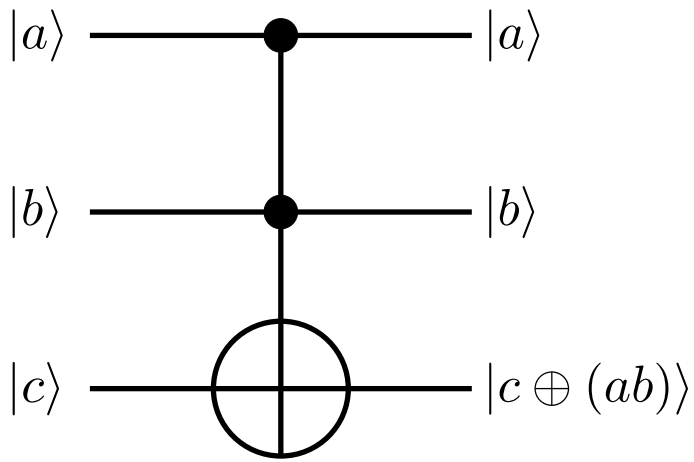
The CNOT gate is a reversible version of `XOR`:



$\|ab\rangle$	$CNOT\|ab\rangle = \|a(b \oplus a)\rangle$
$\|00\rangle$	$\|00\rangle$
$\|01\rangle$	$\|01\rangle$
$\|10\rangle$	$\|11\rangle$
$\|11\rangle$	$\|10\rangle$

Reversible circuits

The Toffoli gate is a reversible version of `AND`:

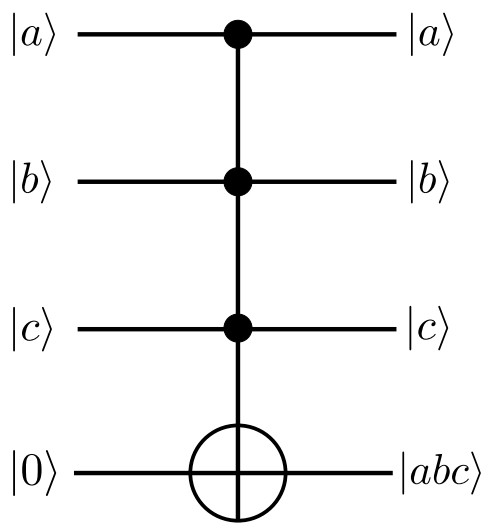


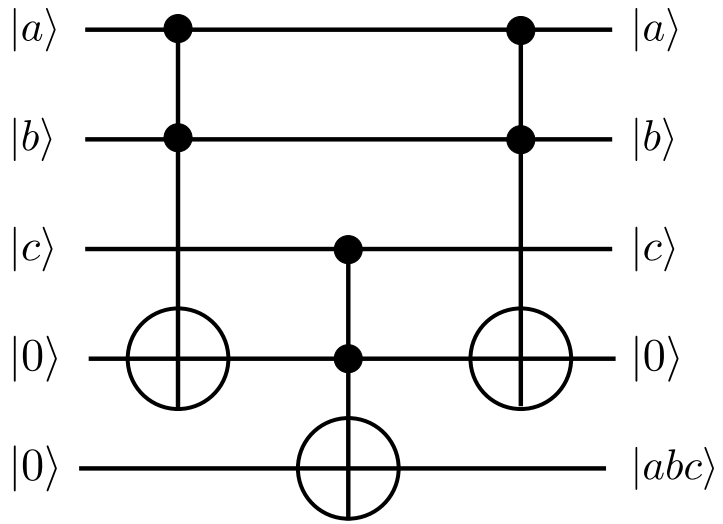
$\lvert abc\rangle$	$TOF\lvert abc\rangle = \lvert ab(c \oplus ab)\rangle$
$\lvert 000\rangle$	$\lvert 000\rangle$
$\lvert 001\rangle$	$\lvert 001\rangle$
$\lvert 010\rangle$	$\lvert 011\rangle$
$\lvert 011\rangle$	$\lvert 010\rangle$
$\lvert 100\rangle$	$\lvert 100\rangle$
$\lvert 101\rangle$	$\lvert 101\rangle$
$\lvert 110\rangle$	$\lvert 111\rangle$
$\lvert 111\rangle$	$\lvert 110\rangle$

Furthermore, the Toffoli is **universal** for reversible Boolean functions, provided there are enough **auxiliary** qubits.

Auxiliary qubits

Auxiliary qubits are like "scratch", or "work" qubits. They start in state $|0\rangle$, and must be returned to state $|0\rangle$, but can be used to store intermediate results in a computation.





Universal gate sets

Last week, we learned that with just

- H and T
- any two of RX, RY, and RZ,

we can implement *any* single-qubit unitary operation up to arbitrary precision.

What about for two qubits?

Universal gate sets

What about for two qubits?

- H, T, and $CNOT$
- any two of RX, RY, RZ, and $CNOT$
- H and TOF

With just 2-3 gates, we can implement *any* two-qubit unitary operation up to arbitrary precision.

What about three or more qubits? (Same thing!)

Universal gate sets

In general, finding such an implementation (*quantum circuit synthesis*, part of the quantum compilation pipeline) is computationally hard.

- sometimes we can do so for small cases (PennyLane has many decompositions pre-programmed)
- sometimes having **auxiliary** qubits around can simplify the decomposition

Multi-qubit measurements

Review: single-qubit measurements

Given a state

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

- the probability of measuring and observing the qubit in state $|0\rangle$ is $|\alpha|^2 = \alpha\alpha^* = |\langle 0|\psi\rangle|^2$
- the probability of measuring and observing the qubit in state $|1\rangle$ is $|\beta|^2 = |\langle 1|\psi\rangle|^2$
- we can measure in different bases by "remapping" those basis states to the computational basis
- we can compute the expectation value of an observable M as $\langle\psi| M|\psi\rangle$ (and estimate it from the results of many samples)

We can do all this in the multi-qubit case as well.

Multi-qubit measurement outcome probabilities

Let

$$|\psi\rangle = \alpha|00\rangle + \beta|01\rangle + \gamma|10\rangle + \delta|11\rangle$$

If we measure in the computational basis, the outcome probabilities are:

- $|\alpha|^2 = |\langle 00|\psi\rangle|^2$ for $|00\rangle$
- $|\beta|^2 = |\langle 01|\psi\rangle|^2$ for $|01\rangle$
- ...

Multi-qubit measurement outcome probabilities

Let

$$|\psi\rangle = \alpha|00\rangle + \beta|01\rangle + \gamma|10\rangle + \delta|11\rangle$$

We can also measure *just one qubit*:

- The probability of the first qubit being in state $|0\rangle$ is $|\alpha|^2 + |\beta|^2$
- The probability of the second qubit being in state $|1\rangle$ is $|\beta|^2 + |\delta|^2$

Multi-qubit measurement outcome probabilities

Let

$$|\psi\rangle = \alpha|00\rangle + \beta|01\rangle + \gamma|10\rangle + \delta|11\rangle$$

We can also measure *just one qubit*:

- The probability of the first qubit being in state $|0\rangle$ is $|\alpha|^2 + |\beta|^2$
- The probability of the second qubit being in state $|1\rangle$ is $|\beta|^2 + |\delta|^2$

Multi-qubit expectation values

Single-qubit observables are 2×2 Hermitian matrices; multi-qubit observables are $2^n \times 2^n$ Hermitian matrices.

Mostly we concern ourselves with observables from the n-qubit Pauli group, $\mathcal{P}_n$:

- I, X, Y, Z for one qubit
- $X \otimes I$, $I \otimes X$, $X \otimes X$, $X \otimes Y$, ... for 2 qubits

4^n possible Pauli operators for n qubits (including the identity).

Why the Pauli group?

Multi-qubit expectation values

You can write *any Hermitian matrix* as a linear combination of Pauli operators (they are a basis). Expectation values are linear.

Example:

$$M = \sum_{i=1}^{4^n} c_i P_i, \quad P_i \in \mathcal{P}_n$$

Then for some state $|\psi\rangle$,

Multi-qubit expectation values

Example: operator $Z \otimes Z$.

Eigenvalues are computational basis states:

$$
\begin{aligned}
(Z \otimes Z)|00\rangle &= |00\rangle \\
(Z \otimes Z)|01\rangle &= -|01\rangle \\
(Z \otimes Z)|10\rangle &= -|10\rangle \\
(Z \otimes Z)|11\rangle &= |11\rangle
\end{aligned}
$$

To compute an expectation value from data:

$$
\langle Z \otimes Z \rangle = \frac{1 \cdot n_1 + (-1) \cdot n_{-1}}{N}
$$

Multi-qubit expectation values

Example: operator $X \otimes I$.

Eigenvalues of X are the $|+\rangle$ and $|-\rangle$ states:

$$
\begin{aligned}
(X \otimes I)|+0\rangle &= |+0\rangle \\
(X \otimes I)|+1\rangle &= |+1\rangle \\
(X \otimes I)|-0\rangle &= -|-0\rangle \\
(X \otimes I)|-1\rangle &= -|-1\rangle
\end{aligned}
$$

Fun fact: All Pauli operators have an equal number of $+1$ *and* -1 *eigenvalues!*

Multi-qubit expectation values

How to compute expectation value of X from data, when we can only measure in the computational basis?

Basis rotation: apply H to first qubit

$$(H \otimes I)(X \otimes I)|+0\rangle = |00\rangle$$
$$(H \otimes I)(X \otimes I)|+1\rangle = |01\rangle$$
$$(H \otimes I)(X \otimes I)|-0\rangle = -|10\rangle$$
$$(H \otimes I)(X \otimes I)|-1\rangle = -|11\rangle$$

When we measure and obtain $|10\rangle$ or $|11\rangle$, we know those correspond to the -1 eigenstates of $X \otimes I$.

Hands-on: multi-qubit expectation values

Multi-qubit expectation values can be created using the `@` symbol:

```
@qml.qnode(dev)
def circuit(x):
    qml.Hadamard(wires=0)
    qml.CRX(x, wires=[0, 1])
    return qml.expval(qml.PauliZ(0) @ qml.PauliZ(1))
```

Hands-on: multi-qubit expectation values

Can also return *multiple* expectation values, if there are no shared qubits.

```
@qml.qnode(dev)
def circuit(x):
    qml.Hadamard(wires=0)
    qml.CRX(x, wires=[0, 1])
    return qml.expval(qml.PauliZ(0)), qml.expval(qml.
                                        PauliZ(1))
```

Superdense coding

Suppose Alice wants to send Bob two classical bits of information, say ‘1’ and ‘0’.

Q1: How many classical bits does she need to send to Bob to do this?

Superdense coding

Suppose Alice wants to send Bob two classical bits of information, say ‘1’ and ‘0’.

Q1: How many classical bits does she need to send to Bob to do this?
A1: 2.

Superdense coding

Suppose Alice wants to send Bob two classical bits of information, say ‘1’ and ‘0’.

Q1: How many classical bits does she need to send to Bob to do this?
A1: 2.

Q2: How many *qubits* does she need to do this?

Superdense coding

Suppose Alice wants to send Bob two classical bits of information, say ‘1’ and ‘0’.

Q1: How many classical bits does she need to send to Bob to do this?
A1: 2.

Q2: How many *qubits* does she need to do this?
A2: Only 1!

Superdense coding

Alice can send 2 bits with a single qubit if her and Bob share a pair of entangled qubits:

$$|\Phi\rangle_{AB} = \frac{1}{\sqrt{2}}\left(|00\rangle + |11\rangle\right) \tag{1}$$

Alice will perform some operations on her qubit depending on which bit she wants to send. Then she will send her qubit to Bob.

Once Bob receives Alice's qubit, he measure both qubits to determine the bits she wanted to send him.

Superdense coding

Alice and Bob start the protocol with this entangled state:

$$|\Phi\rangle_{AB} = \frac{1}{\sqrt{2}}\left(|00\rangle + |11\rangle\right) \tag{2}$$

Next, depending on her bits, Alice performs one of the following operations on her qubit:

$$00 \rightarrow I \tag{3}$$
$$01 \rightarrow X \tag{4}$$
$$10 \rightarrow Z \tag{5}$$
$$11 \rightarrow ZX \tag{6}$$

Superdense coding

What happened to the entangled state?

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

It will transform to:

$$00 \quad \rightarrow_I$$

$$01 \quad \rightarrow_X$$

$$10 \quad \rightarrow_Z$$

$$11 \quad \rightarrow_{ZX}$$

Superdense coding

There is something special about these four states: they are a set of 4 orthonormal states called the *Bell basis*.

$$
\begin{aligned}
|\Phi_0\rangle &= \frac{1}{\sqrt{2}}\left(|00\rangle + |11\rangle\right) \\
|\Phi_1\rangle &= \frac{1}{\sqrt{2}}\left(|10\rangle + |01\rangle\right) \\
|\Phi_2\rangle &= \frac{1}{\sqrt{2}}\left(|00\rangle - |11\rangle\right) \\
|\Phi_3\rangle &= \frac{1}{\sqrt{2}}\left(-|10\rangle + |01\rangle\right)
\end{aligned}
$$

Bob can perform a measurement in it to determine with certainty which state he has, and correspondingly which bits Alice sent him.

Fun with circuits: write a quantum circuit to produce each of these states. See Codebook I.14.

Superdense coding

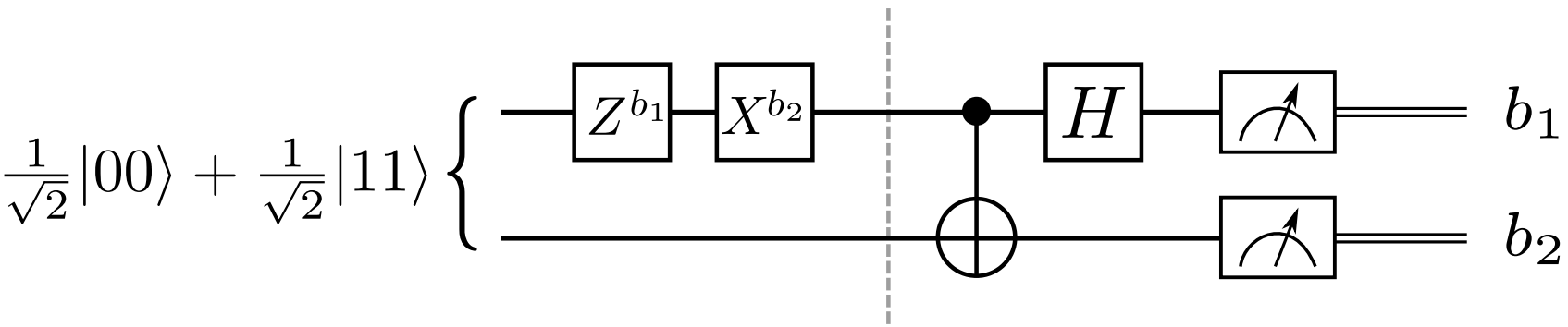
Bob can measure directly in the Bell basis, or he can perform a basis transformation from the Bell basis back to the computational basis:

$$
\begin{aligned}
(H \otimes I)\mathrm{CNOT} \cdot \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) &= |00\rangle \\
(H \otimes I)\mathrm{CNOT} \cdot \frac{1}{\sqrt{2}}(|10\rangle + |01\rangle) &= |01\rangle \\
(H \otimes I)\mathrm{CNOT} \cdot \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle) &= |10\rangle \\
(H \otimes I)\mathrm{CNOT} \cdot \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle) &= |11\rangle
\end{aligned}
$$

Then he can measure in the computational basis to get the bits directly.

Hands-on: superdense coding

Let's go implement it!

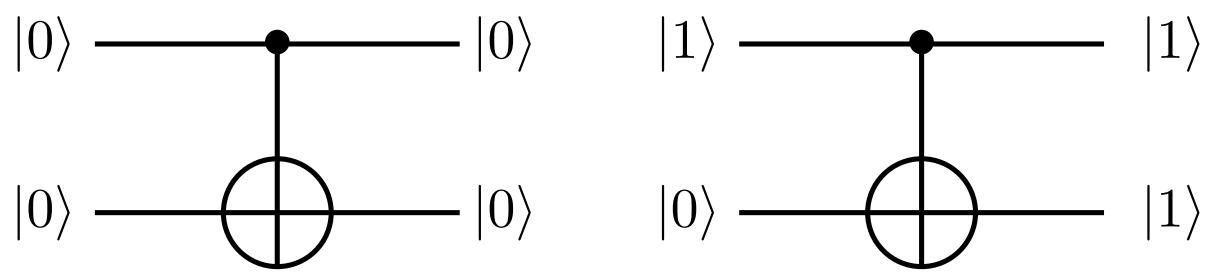


Teleportation

Copying quantum states

Suppose you found a really cool quantum state, and you want to send a copy to a friend. Can you?

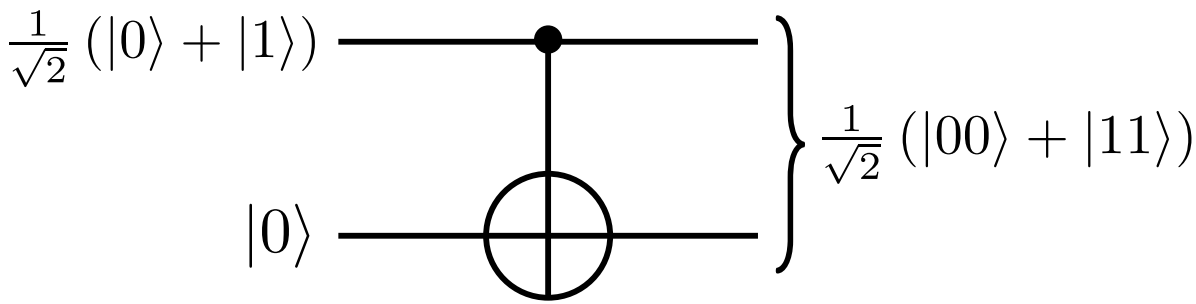
Idea: CNOT sends $|00\rangle$ to $|00\rangle$, and $|10\rangle$ to $|11\rangle$, thus copying the first qubit's state to the second.



Everything is linear, so will this work in general?

Copying quantum states

Very easy to find a state for which this fails:



(Not) copying quantum states

The no-cloning theorem

It is impossible to create a copying circuit that works for arbitrary quantum states.

In other words, there is no circuit that sends

$$|\psi\rangle \otimes |s\rangle \to |\psi\rangle \otimes |\psi\rangle$$

for any arbitrary $|\psi\rangle$.

Proof of the no-cloning theorem

Suppose we want to clone a state $|\psi\rangle$. We want a unitary operation that sends

where $|s\rangle$ is some arbitrary state.

Let's suppose we find one. If our cloning machine is going to be universal, then we must also be able to clone some other state, $|\varphi\rangle$.

Proof of the no-cloning theorem

We purportedly have:

$$\begin{aligned} U(|\psi\rangle \otimes |s\rangle) &= |\psi\rangle \otimes |\psi\rangle \\ U(|\varphi\rangle \otimes |s\rangle) &= |\varphi\rangle \otimes |\varphi\rangle \end{aligned}$$

Take the inner product of the LHS of both equations:

Now take the inner product of the RHS of both equations:

Proof of the no-cloning theorem

For what states does

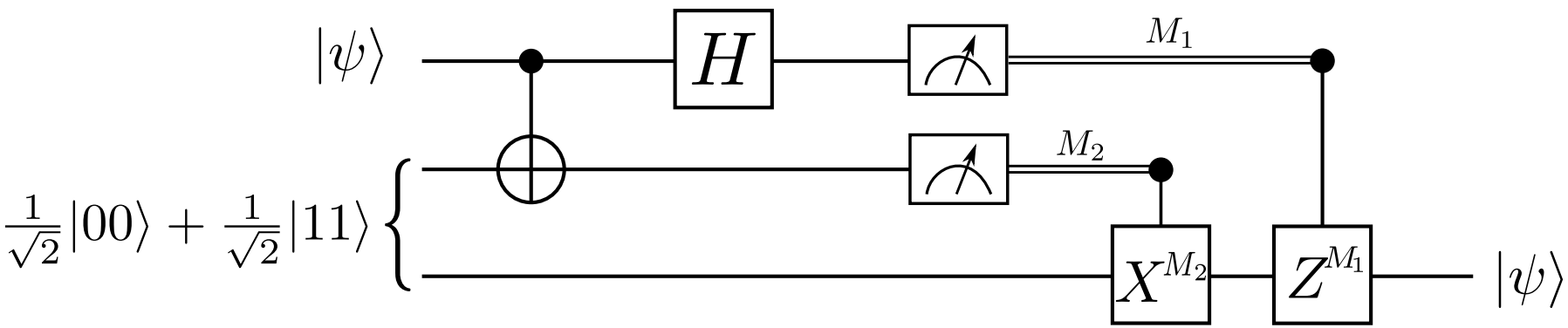
$$(\langle\psi|\varphi\rangle)^2 = \langle\psi|\varphi\rangle$$

Need a complex number that squares to itself... but the only numbers that square to themselves are 0 and 1!

So either the two states are orthogonal, or are just the same state. They can't be arbitrary!

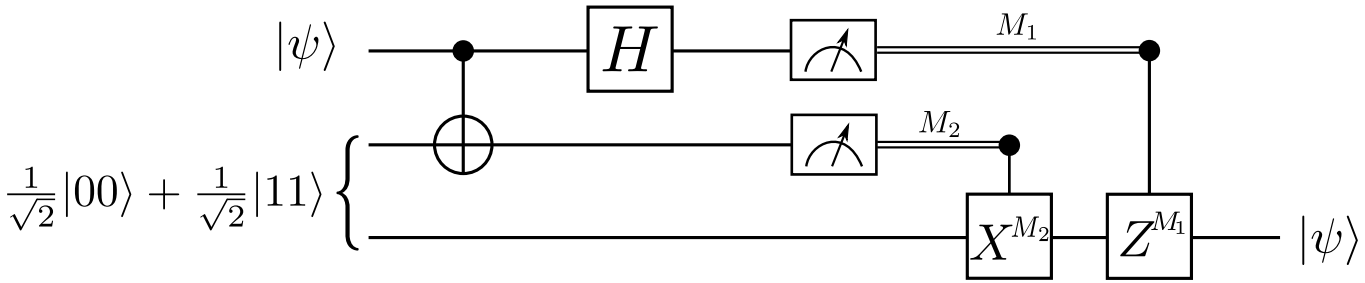
Teleportation

We cannot clone arbitrary qubit states, but we *can* teleport them!



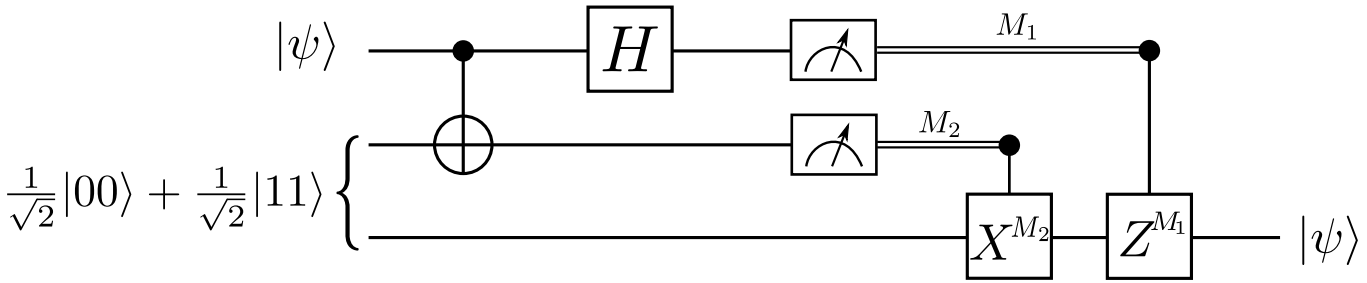
Quantum teleportation: the details

For this protocol, it is extremely valuable to work through it one gate at a time.



Quantum teleportation: the details

For this protocol, it is extremely valuable to work through it one gate at a time.



Quantum teleportation: the details

Before measurements, the combined state of the system is (removing the $\frac{1}{2}$ for readability):

$$\begin{aligned}
|00\rangle &\otimes (\alpha|0\rangle + \beta|1\rangle) + \\
|01\rangle &\otimes (\alpha|1\rangle + \beta|0\rangle) + \\
|10\rangle &\otimes (\alpha|0\rangle - \beta|1\rangle) + \\
|11\rangle &\otimes (\alpha|1\rangle - \beta|0\rangle)
\end{aligned}$$

This is a superposition of 4 distinct terms.

Quantum teleportation: the details

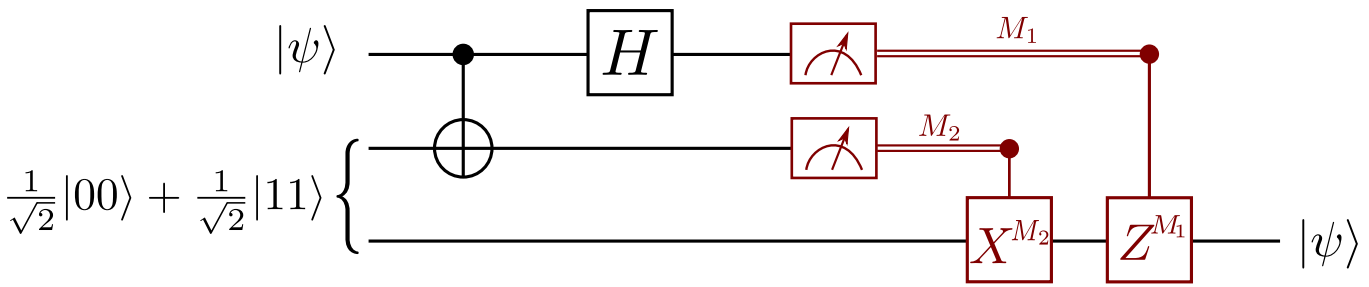
You can see that Bob's state is always some variation on the original state of Alice:

$$
\begin{aligned}
|00\rangle &\otimes (\alpha|0\rangle + \beta|1\rangle) + \\
|01\rangle &\otimes (\alpha|1\rangle + \beta|0\rangle) + \\
|10\rangle &\otimes (\alpha|0\rangle - \beta|1\rangle) + \\
|11\rangle &\otimes (\alpha|1\rangle - \beta|0\rangle)
\end{aligned}
$$

Quantum teleportation: the details

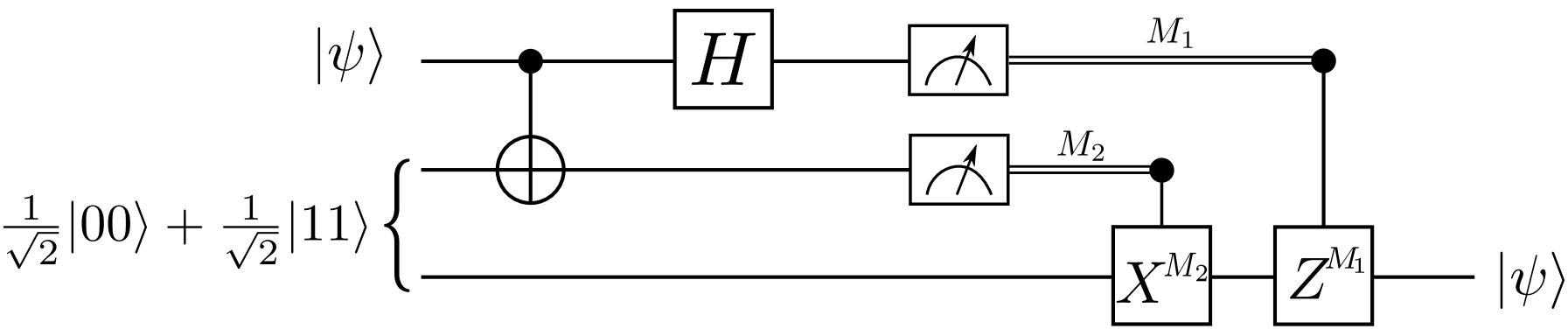
Alice measures in the computational basis and sends her results to Bob. Once Bob knows the results, he knows exactly what term of the superposition they had, and can adjust his state accordingly.

$$
\begin{aligned}
00 &: I(\alpha|0\rangle + \beta|1\rangle) = (\alpha|0\rangle + \beta|1\rangle) \\
01 &: X(\alpha|1\rangle + \beta|0\rangle) = (\alpha|0\rangle + \beta|1\rangle) \\
10 &: Z(\alpha|0\rangle - \beta|1\rangle) = (\alpha|0\rangle + \beta|1\rangle) \\
11 &: ZX(\alpha|1\rangle - \beta|0\rangle) = (\alpha|0\rangle + \beta|1\rangle)
\end{aligned}
$$



Hands-on: teleportation

Need to deal with the fact that we cannot perform classically controlled operations in PennyLane...

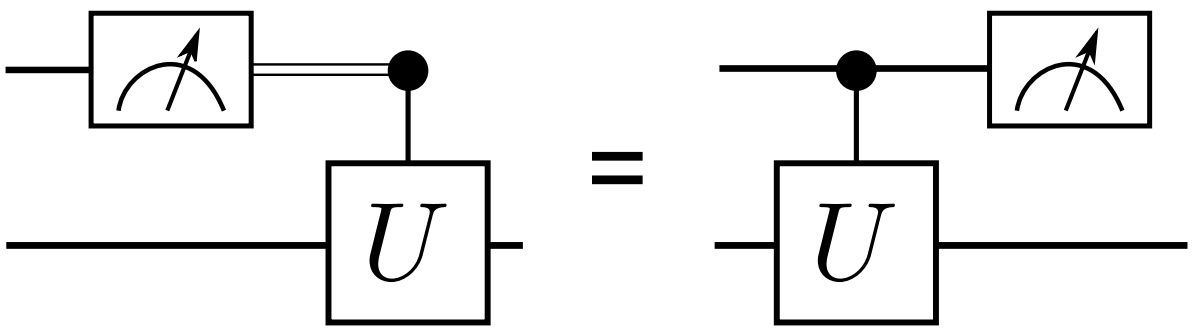


Principle of deferred measurement

Nielsen & Chuang:

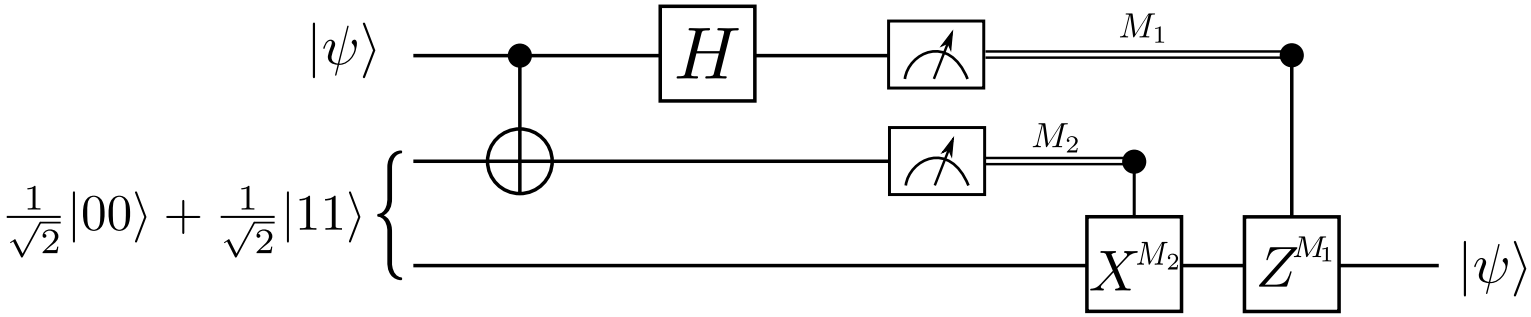
> *"Measurements can always be moved from an intermediate stage of a quantum circuit to the end of the circuit; if the measurement results are used at any stage of the circuit then the classically controlled operations can be replaced by conditional quantum operations."*

Basically,

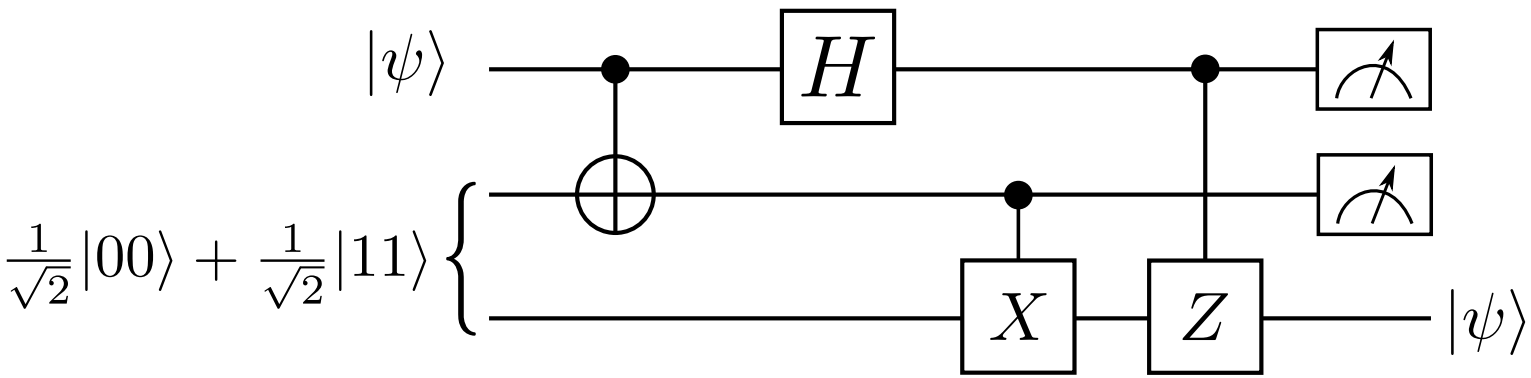


Principle of deferred measurement

Means we can replace our original circuit:



with



Let's implement it!

Recap

- Make any gate a controlled gate
- Perform measurements on multiple qubits
- Implement the superdense coding protocol
- Teleport a qubit

What topics did you find unclear today?

Next time

Content:

- Oracles and black boxes
- Primer on quantum complexity theory
- The Deutsch-Jozsa algorithm

Action items:

1. Continue with Assignment 1 (you can do all the problems now)

Recommended reading:

- Codebook node I.15 (we have covered all of I)
- Nielsen & Chuang 4.4