Intro to Quantum Computing

TRIUMF co-op student seminar #1

Olivia Di Matteo
Quantum Information Science Associate, TRIUMF

https://github.com/glassnotes/Intro-QC-TRIUMF

5 February 2020

Overview

Learning goals

Quantum computers will be a very important computational tool in the future. Now is the time to learn how to use them!

At a conceptual level, you'll be able to...

- Explain the motivation behind building quantum computers
- Describe the principles that give quantum computers their "source of power"
- Explain the idea of quantum advantage
- List the major technological players, the main physical implementations, and the successes and challenges of current-generation machines

Learning goals

Using a mixture of theory and hands-on activities, you'll...

- Perform computations and measurements on a single qubit
- Express quantum computations as quantum circuits
- Perform computations on multiple qubits
- Implement a circuit that *teleports* a single-qubit state

Motivation

Physical limitations

'Classical' computers are made more powerful by

- making smaller transistors
- putting more transistors onto a single chip
- using multiple processors in parallel

Physical limitations

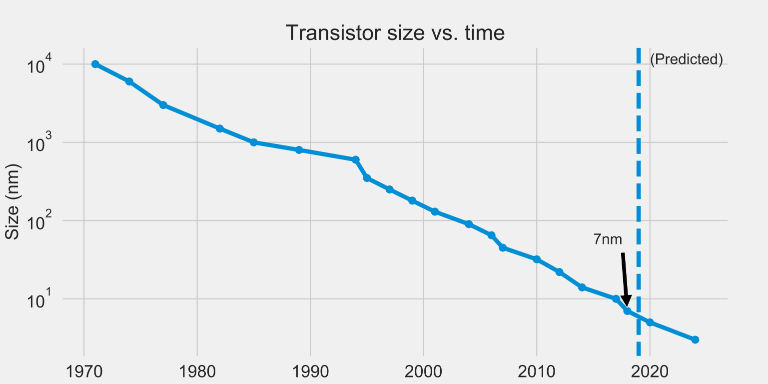
'Classical' computers are made more powerful by

- making smaller transistors
- putting more transistors onto a single chip
- using multiple processors in parallel

Moore's law suggests that every 18 months - 2 years the number of transistors on a chip will double, while the cost is halved.

Why quantum computing?

But we are approaching the physical limits of how small a transistor we can put on a chip before quantum effects (tunneling) will become a problem.



Data source: `https://en.wikipedia.org/wiki/Semiconductor_device_fabrication`

Why quantum computing?

SEMICONDUCTOR ENGINEERING

Home > Manufacturing, Packaging & Materials > Quantum Effects At 7/5nm And Beyond

MANUFACTURING, PACKAGING & MATERIALS

Quantum Effects At 7/5nm And Beyond

At future nodes there are some unexpected behaviors. What to do about them isn't always clear.

MAY 23RD, 2018 - BY: **ED SPERLING**

Manufacturing of devices with 7nm chips began late in 2018.

Mass production of 5nm chips to begin in 2020 (limited production began in 2019).

Why quantum computing?

SEMICONDUCTOR ENGINEERING

Home > **Manufacturing, Packaging & Materials** > Quantum Effects At 7/5nm And Beyond

MANUFACTURING, PACKAGING & MATERIALS

Quantum Effects At 7/5nm And Beyond

At future nodes there are some unexpected behaviors. What to do about them isn't always clear.

MAY 23RD, 2018 - BY: **ED SPERLING**

Manufacturing of devices with 7nm chips began late in 2018.

Mass production of 5nm chips to begin in 2020 (limited production began in 2019).

Chips with transistors smaller than 7nm have required new and costly fabrication techniques; unclear whether anything smaller than 3nm is viable.

Computational limitations

Some problems will take an intractable amount of time to run on classical computers.

Parallelization can help, but we still cannot fully counteract the exponential complexity of some problems.

Why quantum computing?

Computational limitations

Some problems will take an intractable amount of time to run on classical computers.

Parallelization can help, but we still cannot fully counteract the exponential complexity of some problems.

Sometimes that's a good thing:

- cryptographic infrastructure is built on such mathematically hard problems

Why quantum computing?

But usually it just prevents us from doing interesting things:

- solving complex optimization problems
- simulation of molecules and quantum systems
- searching large spaces
- machine learning with large amounts of data

What is quantum computing?

Quantum computation is the manipulation of the state of a physical quantum system in order to solve a problem.

What is quantum computing?

Quantum computation is the manipulation of the state of a physical quantum system in order to solve a problem.

Many quantum algorithms use 2-level quantum systems called *qubits* to solve problems more efficiently than the best-known classical algorithms:

- Integer factorization *(Shor's algorithm)*
- Searching large configuration spaces *(Grover's algorithm)*
- Simulating quantum systems *(Hamiltonian simulation)*
- Linear algebra *(e.g. HHL algorithm)*
- Machine learning

What's so good about qubits?

- The size of the mathematical space grows *exponentially* in the number of qubits

What's so good about qubits?

- The size of the mathematical space grows *exponentially* in the number of qubits
- We can make linear combinations of all possible qubit states, i.e. we can put them in *superposition*

What's so good about qubits?

- The size of the mathematical space grows *exponentially* in the number of qubits
- We can make linear combinations of all possible qubit states, i.e. we can put them in *superposition*
- We can *entangle* multiple qubits, and use these states as a resource in many algorithms

Single-qubit systems

From bits to qubit I: bits

All computation in our computers today is done with bits.

The state* of a bit is *either* 0 or 1.

A set of n bits is represented by n real numbers (0s and 1s).

- 0

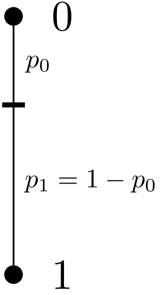
- 1

*Physically this is represented by some voltage and its state depends on whether or not that voltage is above/below a threshold value

From bits to qubit II: probabilistic bits

The value of a bit of information can be governed by a probability distribution.

For example, the outcome of a coin flip is a single bit of information, but before it is flipped it is probabilistic bit with both outcomes equally likely.

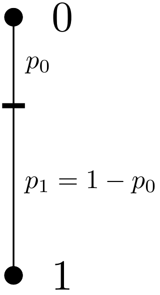


We can assign a probabilistic bit a state based on its probability distribution:

$$\psi = \begin{pmatrix} p_0 \\ p_1 \end{pmatrix}$$

where p_0 and p_1 are real numbers and we must have

$$p_0 + p_1 = 1.$$



From bits to qubit II: probabilistic bits

We can transform one probability distribution to another using a stochastic matrix,

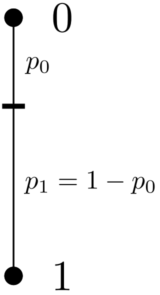
$$A = \begin{pmatrix} a_{00} & a_{01} \\ a_{10} & a_{11} \end{pmatrix}$$

where all $a_{ij} \in [0, 1]$, and the sum of every row is 1.

Then

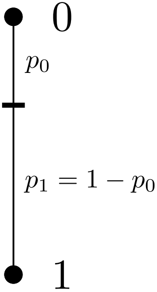
$$\begin{pmatrix} q_0 \\ q_1 \end{pmatrix} = A \begin{pmatrix} p_0 \\ p_1 \end{pmatrix}$$

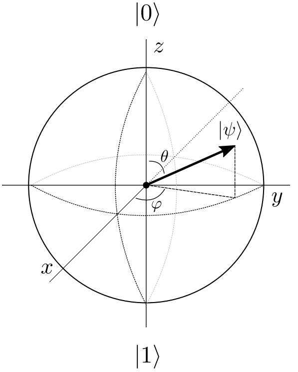
and $q_0 + q_1 = 1$.



From bits to qubit III: qubits

Extension of probabilistic bits to 2-level quantum systems:





Extension of probabilistic bits to 2-level quantum systems.

Instead of

$$\psi = \begin{pmatrix} p_0 \\ p_1 \end{pmatrix}$$

where $p_0, p_1 \in \mathbb{R}$ and $p_0 + p_1 = 1$, we have

$$|\psi\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}$$

where $\alpha, \beta \in \mathbb{C}$, and $|\alpha|^2 + |\beta|^2 = 1$.

A qubit state is a unit vector that lives in a 2-dimensional complex vector space called *Hilbert space*.

A qubit state is a unit vector that lives in a 2-dimensional complex vector space called *Hilbert space*.

Qubit states are written as a linear combination, or *superposition*, of an *orthonormal basis* of the Hilbert space. The most commonly used one is the **computational basis**:

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

Then,

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \quad \alpha, \beta \in \mathbb{C},$$

where unit length is ensured by having $|\alpha|^2 + |\beta|^2 = 1$.

Mathematical representation of qubits

The complex parameters α and β are called *amplitudes*.

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle = \alpha \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \beta \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

If we measure this qubit, we will find it in

- state $|0\rangle$ with probability $|\alpha|^2$,
- state $|1\rangle$ with probability $|\beta|^2$.

(Hence the need for the restriction $|\alpha|^2 + |\beta|^2 = 1.$)

Such a measurement is *destructive* - afterwards, the qubit stays in the state in which we observed it.

We will talk more about measurements later today.

Using qubits for computation

The amplitudes are the key here - when we measure the qubits at the end of our computation, they are what determine the outcome frequencies. We should make them meaningful to the problem.

But for this we need a way of:

1. putting qubits into superposition

Using qubits for computation

The amplitudes are the key here - when we measure the qubits at the end of our computation, they are what determine the outcome frequencies. We should make them meaningful to the problem.

But for this we need a way of:

1. putting qubits into superposition
2. manipulating the qubit to get the amplitudes that we want

Using qubits for computation

The amplitudes are the key here - when we measure the qubits at the end of our computation, they are what determine the outcome frequencies. We should make them meaningful to the problem.

But for this we need a way of:

1. putting qubits into superposition
2. manipulating the qubit to get the amplitudes that we want
3. combining multiple qubits

Using qubits for computation

The amplitudes are the key here - when we measure the qubits at the end of our computation, they are what determine the outcome frequencies. We should make them meaningful to the problem.

But for this we need a way of:

1. putting qubits into superposition
2. manipulating the qubit to get the amplitudes that we want
3. combining multiple qubits
4. measuring the qubits to get the answer

The amplitudes are the key here - when we measure the qubits at the end of our computation, they are what determine the outcome frequencies. We should make them meaningful to the problem.

But for this we need a way of:

1. putting qubits into superposition
2. manipulating the qubit to get the amplitudes that we want
3. combining multiple qubits
4. measuring the qubits to get the answer

How do we (mathematically) perform operations on qubits?

Single-qubit states are manipulated by 2×2 *unitary* matrices. U is a unitary matrix if

$$U^\dagger U = UU^\dagger = \mathbb{1}$$

Unitary operations *preserve the normalization of qubit states*. If

$$U(\alpha|0\rangle + \beta|1\rangle) = \alpha'|0\rangle + \beta'|1\rangle,$$

then

$$|\alpha'|^2 + |\beta'|^2 = |\alpha|^2 + |\beta|^2 = 1$$

The principle of superposition

Unitary operations act *linearly* on superpositions:

$$U(\alpha|0\rangle + \beta|1\rangle) = \alpha U|0\rangle + \beta U|1\rangle$$

This is a key contributor to the power of quantum computing!

Creating a superposition: the Hadamard gate

Perhaps the most important unitary in quantum computing is the Hadamard gate:

$$H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

H creates a *uniform superposition* of computational basis states:

$$\begin{aligned} H|0\rangle &= \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = |+\rangle \\ H|1\rangle &= \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = |-\rangle \end{aligned}$$

...and also undoes it:

$$\begin{aligned} H|+\rangle &= |0\rangle \\ H|-\rangle &= |1\rangle \end{aligned}$$

Single-qubit gates: Paulis

You are probably familiar with the Pauli operators, X, Y, and Z:

$$
\begin{aligned}
X &= \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \\
Z &= \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \\
Y = iZX &= \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}
\end{aligned}
$$

These are unitary, and can do some very useful things to qubit states.

Single-qubit gates: Paulis

X is called the **bit flip** operation:

$$\begin{aligned} X|0\rangle &= |1\rangle, \\ X|1\rangle &= |0\rangle \end{aligned}$$

Z is called the **phase flip** operation:

$$\begin{aligned} Z|0\rangle &= |0\rangle, \\ Z|1\rangle &= -|1\rangle \end{aligned}$$

Single-qubit gates: applying sequences of gates

We apply *products* of single-qubit operations to represent performing gates in sequence. For example,

$$\begin{aligned} XZH|0\rangle &= XZ\left(\frac{|0\rangle}{\sqrt{2}} + \frac{|1\rangle}{\sqrt{2}}\right) \\ &= X\left(\frac{|0\rangle}{\sqrt{2}} - \frac{|1\rangle}{\sqrt{2}}\right) \\ &= \frac{|1\rangle}{\sqrt{2}} - \frac{|0\rangle}{\sqrt{2}} \end{aligned}$$

Products are applied from right to left.

Single-qubit gates: rotations

X, Y, and Z are special cases of more general qubit rotations:

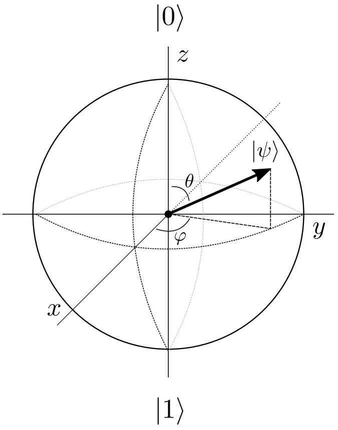
$$
\begin{aligned}
R_x(\theta) &= \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & -i\sin\left(\frac{\theta}{2}\right) \\ -i\sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{pmatrix} \\
R_y(\theta) &= \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & -\sin\left(\frac{\theta}{2}\right) \\ \sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{pmatrix} \\
R_z(\theta) &= \begin{pmatrix} e^{-i\frac{\theta}{2}} & 0 \\ 0 & e^{i\frac{\theta}{2}} \end{pmatrix}
\end{aligned}
$$

$X = R_x(\pi),\ \ Y = R_y(\pi),\ \ Z = R_z(\pi).$

R_x and R_y can put qubits into *non-uniform* superpositions.

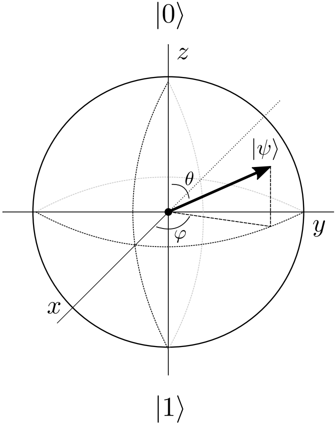
... wait. Rotations? Rotations around *what*?

Single-qubit ket states can be represented on the surface of a sphere of radius 1 called the *Bloch sphere*.



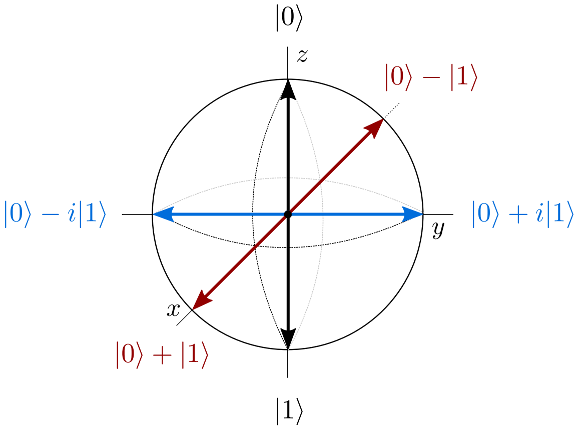
States can be plotted on the sphere using a parameterized form:

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

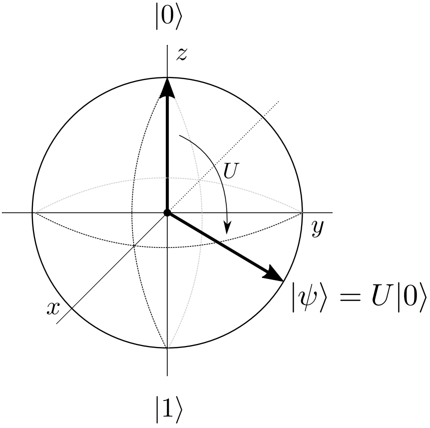


Visualizing a qubit: the Bloch sphere

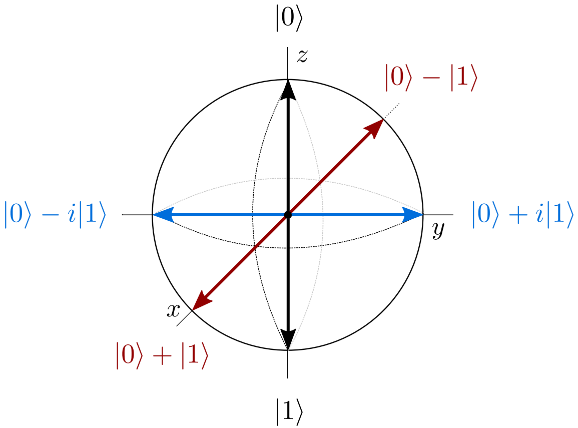
The axis points correspond to the *eigenstates* of σ_x, σ_y, and σ_z.



Unitary matrices rotate state vectors around the Bloch sphere



Operations on the Bloch sphere



Single-qubit gates: complex phase gates

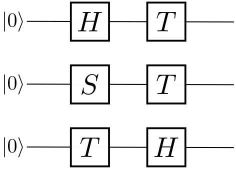
There are two more very important single-qubit gates:

Gate	**Unitary**	**Action**
Phase gate $S = \sqrt{Z}$	$\begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$	$1/4$ turn around z-axis
T gate, $T = \sqrt{S}$	$\begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}$	$1/8$ turn around z-axis

These gates change the *relative phase* between $|0\rangle$ and $|1\rangle$. They don't affect the magnitudes of the amplitudes, but we will see that they *do* affect the measurement statistics!

Quantum circuits

That is a lot of matrices and vectors - it is going to get even more tedious when we bring in more qubits.
Thankfully we have *quantum circuits*.



Gates are applied from left to right, e.g. apply H on qubit 1, then apply T.

Unitaries are applied from right to left, e.g. qubit 1 gets hit with $U = TH$.

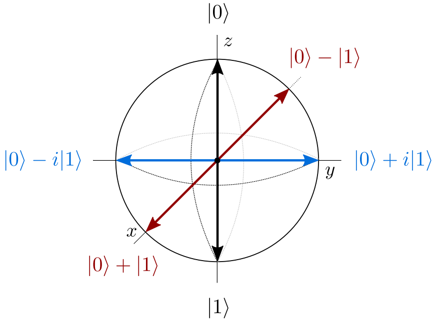
Single-qubit operations: hands-on with Quirk

Navigate to:

`https://algassert.com/quirk`

Single-qubit operations: hands-on with Quirk

Task 1: Using only H and Z gates, implement an X gate



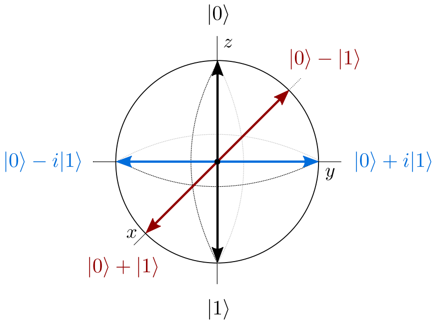
$$
\begin{aligned}
X|0\rangle &= |1\rangle \\
X|1\rangle &= |0\rangle \\
Z|0\rangle &= |0\rangle \\
Z|1\rangle &= -|1\rangle \\
H|0\rangle &= \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \\
H|1\rangle &= \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)
\end{aligned}
$$

Note: any gates you add get added to the URL, so to save a circuit, you can just bookmark the webpage.

Single-qubit operations: hands-on with Quirk

Task 2: Starting from $|0\rangle$, prepare the state

$$|\psi\rangle = \frac{1}{2}|0\rangle + \frac{\sqrt{3}i}{2}|1\rangle$$



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

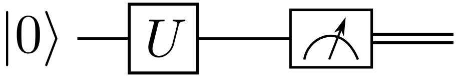
$$R_y(\theta) = \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & -\sin\left(\frac{\theta}{2}\right) \\ \sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{pmatrix}$$

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

Measurement

We know now that *unitary operations* can be used to manipulate our qubits to perform a computation. But after we're done computing, how do we get the answer? We need to measure our system.

A measurement in a circuit is represented by a box with a dial:



The two wires coming out of it indicate a *classical bit* - the outcome of the measurement is not a qubit, it's either 0 or 1!

We need a mathematical formalism for measuring qubits, to see what the state system is in after the computation.

If we measure a qubit prepared in state $|\psi\rangle$, the probability of observing it in state $|\varphi\rangle$ is

$$\text{Pr(outcome i)} = |\langle\varphi|\psi\rangle|^2$$

After the measurement the qubit will be left in state $|\varphi\rangle$[1].

[1] Actually things are a bit more subtle than this; for a good overview of projective measurements, see `https://www.people.vcu.edu/~sgharibian/courses/CMSC491/notes/Lecture%203%20-%20Measurement.pdf`

Measurement

Measurement is performed with respect to a basis, for example, the computational basis $\{|0\rangle, |1\rangle\}$

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

Then if we measure $|\psi\rangle$, we will observe the system in state $|0\rangle$ or state $|1\rangle$ with probability

$$\begin{aligned} \Pr(0) &= |\langle 0|\psi\rangle|^2 = |\alpha|^2 \\ \Pr(1) &= |\langle 1|\psi\rangle|^2 = |\beta|^2 \end{aligned}$$

Measurement in other bases

We can measure in any orthonormal basis by applying a suitable unitary transformation to the computational basis vectors.

Example: measuring in the *Hadamard basis*:

$$\begin{aligned} |+\rangle &= H|0\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right) \\ |-\rangle &= H|1\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right) \end{aligned}$$

Measurement in other bases

Then, for $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$,

$$\begin{aligned}
\Pr(+) &= |\langle +|\psi\rangle|^2 \\
&= \frac{1}{2}|\alpha\langle 0|0\rangle + \alpha\langle 0|1\rangle + \beta\langle 1|0\rangle + \beta\langle 1|1\rangle|^2 \\
&= \frac{1}{2}|\alpha + \beta|^2 \\
\Pr(-) &= \frac{1}{2}|\alpha - \beta|^2
\end{aligned}$$

Measurement in other bases

Why would we want to measure in different bases?

Example

Consider $|+\rangle$ and $|-\rangle$.

If we measure in the computational basis, for both states we will get 0 with probability $1/2$ and also 1 with probability $1/2$. It's impossible to know which state we have!

But if we measure in the Hadamard basis, we will get either *only* $+$ or *only* $-$.

The measurement statistics change depending on which basis we measure in! Tomorrow, we will see an example (quantum teleporation) of how this is useful.

Review

We talked about:

- Why quantum computing will be needed in the future
- What qubits are, and how they are represented mathematically
- Common single-qubit gates
- Measurement of single-qubit systems

Next time

- Multi-qubit systems and entanglement
- Quantum teleportation
- Quantum advantage
- Overview of current-generation quantum hardware

For more information, advanced topics, and references, check out my introductory QC lecture notes on Github:

`https://github.com/glassnotes/Intro-QC-TRIUMF`

I will post the slides from today there as well.