

Week 6: Quantum Circuits and Deutsch's Algorithm

COMS 4281 (Fall 2025)

- No-cloning theorem
- Quantum teleportation
- Measurements in other bases

Heisenberg Uncertainty Principle

Heisenberg Uncertainty Principle

Popular science version: can't exactly know both the position and momentum of a particle simultaneously.



Heisenberg Uncertainty Principle

In quantum information theory terms: it is not possible for a qubit $|\psi\rangle \in \mathbb{C}^2$ to be simultaneously determined in both the **standard** basis and the **diagonal basis**.

Heisenberg Uncertainty Principle

In quantum information theory terms: it is not possible for a qubit $|\psi\rangle \in \mathbb{C}^2$ to be simultaneously determined in both the **standard** basis and the **diagonal basis**.

In other words, if measuring $|\psi\rangle$ in standard basis yields a deterministic outcome, then it **cannot** have a deterministic outcome if measured according to diagonal basis.

Heisenberg Uncertainty Principle

Quantitatively, let:

1. $p_0 = |\langle 0|\psi\rangle|^2$ and $p_1 = |\langle 1|\psi\rangle|^2$ be probability of outcomes in standard basis.

Heisenberg Uncertainty Principle

Quantitatively, let:

1. $p_0 = |\langle 0|\psi\rangle|^2$ and $p_1 = |\langle 1|\psi\rangle|^2$ be probability of outcomes in standard basis.
2. $p_+ = |\langle +|\psi\rangle|^2$ and $p_- = |\langle -|\psi\rangle|^2$ be probability of outcomes in Hadamard basis.

Heisenberg Uncertainty Principle

Quantitatively, let:

1. $p_0 = |\langle 0|\psi\rangle|^2$ and $p_1 = |\langle 1|\psi\rangle|^2$ be probability of outcomes in standard basis.
2. $p_+ = |\langle +|\psi\rangle|^2$ and $p_- = |\langle -|\psi\rangle|^2$ be probability of outcomes in Hadamard basis.

Then a version of Heisenberg's uncertainty principle says

$$(p_0 - p_1)^2 + (p_+ - p_-)^2 \leq 1$$

Heisenberg Uncertainty Principle

Important point: it's not about what happens if you **sequentially** measure the state $|\psi\rangle$ (what happens then?).

Heisenberg Uncertainty Principle

Important point: it's not about what happens if you **sequentially** measure the state $|\psi\rangle$ (what happens then?).

It's reasoning about **counterfactual scenarios**: measuring $|\psi\rangle$ in the standard basis, **or** measuring $|\psi\rangle$ in the diagonal basis.

Heisenberg Uncertainty Principle

We say that the standard basis and diagonal basis are **incompatible** or **complementary**.

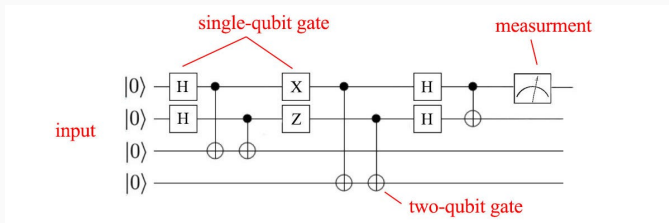
In quantum physics, the position and momentum of a particle correspond to incompatible measurements!

Quantum Computation

What is quantum computation?

It's whatever you can do with a quantum circuit, where (typically):

1. The input qubits start in the $|0\rangle$ state.
2. A sequence of single and two-qubit gates drawn from a **Gate Set**
3. Measurements (usually at the end)



What single- and two qubit gates are allowed in a quantum circuit?

What single- and two qubit gates are allowed in a quantum circuit?

It depends on the hardware!

What single- and two qubit gates are allowed in a quantum circuit?

It depends on the hardware!

A gate set $\mathcal{G}$ is called **universal** if, for any unitary U (which may act on many qubits), one can construct a circuit using gates from $\mathcal{G}$ to *approximate* U .

Notion of approximation

Definition: We say that unitary U ϵ -**approximates** V if for all quantum states $|\psi\rangle$,

$$\left\| U|\psi\rangle - V|\psi\rangle \right\| \leq \epsilon.$$

A circuit C without measurements corresponds to some unitary, so it is meaningful to say that a circuit C approximates a unitary matrix.

Universal gate sets

A **continuous** universal gate set: Rotations

$$\boxed{R_X(\theta)} = \begin{pmatrix} \cos \theta & -i \sin \theta \\ -i \sin \theta & \cos \theta \end{pmatrix} \quad \boxed{R_Y(\theta)} = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix}$$

$$\boxed{R_Z(\theta)} = \begin{pmatrix} e^{-i\theta} & 0 \\ 0 & e^{i\theta} \end{pmatrix}$$

over all $0 \leq \theta \leq 2\pi$ plus Phase shift

$$\boxed{P(\varphi)} = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\varphi} \end{pmatrix}$$

over all $0 \leq \varphi \leq 2\pi$ plus CNOT

Universal gate sets

A **discrete, finite** universal gate set:

$$\boxed{H} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \quad \boxed{S} = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \quad \boxed{\text{CNOT}}$$

and

$$\boxed{T} = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}$$

Universal gate sets

Using a universal gate set $\mathcal{G}$ means, in principle we don't have to worry about what gates are used by a given quantum circuit C .

Universal gate sets

Using a universal gate set $\mathcal{G}$ means, in principle we don't have to worry about what gates are used by a given quantum circuit C .

That's because there is a way to **compile** the circuit C into another circuit C' that uses only gates from $\mathcal{G}$, and C *approximates* C' .

Universal gate sets

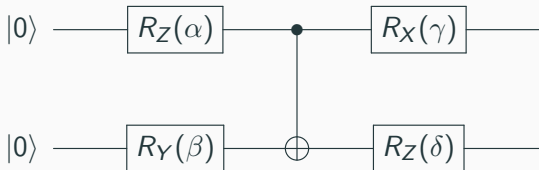
Using a universal gate set $\mathcal{G}$ means, in principle we don't have to worry about what gates are used by a given quantum circuit C .

That's because there is a way to **compile** the circuit C into another circuit C' that uses only gates from $\mathcal{G}$, and C *approximates* C' .

Ideally, we'd like the compiled circuit C' to be not much bigger than C .

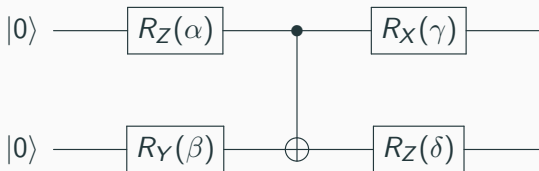
Compiling C into C'

Example: a circuit C with arbitrary rotations

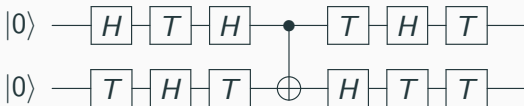


Compiling C into C'

Example: a circuit C with arbitrary rotations



Compiled circuit C' over $\{CNOT, H, T\}$



Solovay-Kitaev theorem

Let $\mathcal{G}$ be a **fixed, finite universal gate set** for one qubit, satisfying suitable technical conditions.

Solovay-Kitaev theorem

Let $\mathcal{G}$ be a **fixed, finite universal gate set** for one qubit, satisfying suitable technical conditions.

For any single-qubit unitary U and $0 < \epsilon < 1/2$, there is a sequence of

$$O\left(\log^c \frac{1}{\epsilon}\right) \text{ gates from } \mathcal{G}$$

whose product ϵ -approximates U (up to a global phase), where c is a fixed constant.

Solovay-Kitaev theorem

Let $\mathcal{G}$ be a **fixed, finite universal gate set** for one qubit, satisfying suitable technical conditions.

For any single-qubit unitary U and $0 < \epsilon < 1/2$, there is a sequence of

$$O\left(\log^c \frac{1}{\epsilon}\right) \text{ gates from } \mathcal{G}$$

whose product ϵ -approximates U (up to a global phase), where c is a fixed constant.

The sequence can be found **efficiently** by a classical algorithm.

Key point: the cost grows polynomially in $\log(1/\epsilon)$, not in $1/\epsilon$.

Consequence: efficient circuit compilation

Let C contain L single- and two-qubit gates. We can compile it into C' using only

$$\{H, T, \text{CNOT}\}.$$

Consequence: efficient circuit compilation

Let C contain L single- and two-qubit gates. We can compile it into C' using only

$$\{H, T, \text{CNOT}\}.$$

For total error at most ϵ (up to global phase), it suffices to use

$$O\left(L \log^c \frac{L}{\epsilon}\right) \text{ gates.}$$

Approximate each original gate to error about ϵ/L .

Consequence: efficient circuit compilation

Let C contain L single- and two-qubit gates. We can compile it into C' using only

$$\{H, T, \text{CNOT}\}.$$

For total error at most ϵ (up to global phase), it suffices to use

$$O\left(L \log^c \frac{L}{\epsilon}\right) \text{ gates.}$$

Approximate each original gate to error about ϵ/L .

Takeaway: Design algorithms using convenient gates, then compile them into a fixed universal gate set.

Quantum circuit synthesis

In quantum computing, we often want to come up with a quantum circuit C to (approximately) implement some unitary U . Hopefully, the circuit C is not too large! This is called **quantum circuit synthesis**.

Quantum circuit synthesis

In quantum computing, we often want to come up with a quantum circuit C to (approximately) implement some unitary U . Hopefully, the circuit C is not too large! This is called **quantum circuit synthesis**.

How many single- and two-qubit gates are needed to build a circuit C that approximates a given unitary U ?

Quantum circuit synthesis

In quantum computing, we often want to come up with a quantum circuit C to (approximately) implement some unitary U . Hopefully, the circuit C is not too large! This is called **quantum circuit synthesis**.

How many single- and two-qubit gates are needed to build a circuit C that approximates a given unitary U ?

In general, at least 4^n !

Our first quantum algorithm

Deutsch's problem

Given oracle access to a boolean function $f : \{0, 1\} \rightarrow \{0, 1\}$, decide whether $f(0) = f(1)$ or $f(0) \neq f(1)$.

Deutsch's problem

Given oracle access to a boolean function $f : \{0, 1\} \rightarrow \{0, 1\}$, decide whether $f(0) = f(1)$ or $f(0) \neq f(1)$.

Oracle access to a function f means that the computer can only access it as a **black-box**, i.e., query some inputs, and get the corresponding outputs. The computer cannot access how the black box is implemented.

Deutsch's problem

Given oracle access to a boolean function $f : \{0, 1\} \rightarrow \{0, 1\}$, decide whether $f(0) = f(1)$ or $f(0) \neq f(1)$.

Oracle access to a function f means that the computer can only access it as a **black-box**, i.e., query some inputs, and get the corresponding outputs. The computer cannot access how the black box is implemented.

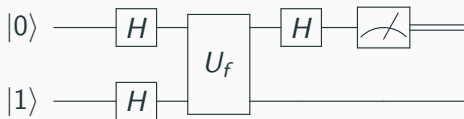
Claim: Any classical algorithm that solves the Deutsch problem must make 2 queries to f .

Quantum algorithms can access a black-box function f through a unitary U_f corresponding to the **reversible** version of f . For $f : \{0, 1\} \rightarrow \{0, 1\}$, this is a two-qubit unitary

$$U_f |x, b\rangle = |x, b \oplus f(x)\rangle.$$

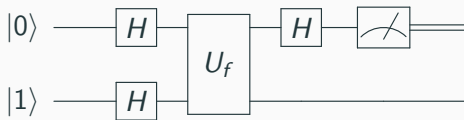
A quantum circuit that wants to access f will simply call U_f just like any other two-qubit gate.

Deutsch's algorithm



This quantum algorithm solves the Deutsch problem with **one** call to U_f .

Deutsch's algorithm



This quantum algorithm solves the Deutsch problem with **one** call to U_f .

Let's do this on the board!

The algorithm evaluates the function f in **superposition**. This seems to give a 2x speedup!

Is this cheating? Maybe the "quantum access" is just really making multiple classical queries under the hood?

Observation: the qubit storing the answer at the end corresponds to the *input wire* of the oracle U_f . We don't care about the output wire!

This is a common feature in many quantum algorithms with exponential speedup.