

Lecture 5

Introduction to quantum algorithms

Luís Soares Barbosa
www.di.uminho.pt/~lsb/



Universidade do Minho



INESCTEC



UNU
EGOV

Quantum Data Science
Universidade do Minho
2026-2027

Recall: A model for quantum computation

States

State of n -qubits encoded as a **unit** vector

$$|u\rangle \in \underbrace{\mathbb{C}^2 \otimes \dots \otimes \mathbb{C}^2}_{n \text{ times}} \cong \mathbb{C}^{2^n}$$

A vector cell is no more a real value in $[0, 1]$, but a **complex** c such that $|c|^2 \in [0, 1]$.

This model expresses a fundamental **physical** concept in quantum mechanics: **interference** — complex numbers may **cancel** each other out when added.

Recall: A model for quantum computation

Dynamics

n -qubit operation encoded as a **unitary transformation**

$$\mathbb{C}^{2^n} \longrightarrow \mathbb{C}^{2^n}$$

i.e. a linear map that preserves inner products, thus norms.

Recall that the norm squared of a unitary matrix forms a double stochastic one.

Recall: A model for quantum computation

Evolution: computed through matrix multiplication with a vector $|u\rangle$ of current **amplitudes** (**wave function**)

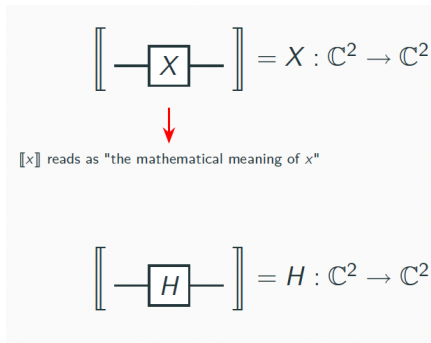
$$M|u\rangle \text{ (next state)}$$

Measurement: **configuration i is observed with probability $|\alpha_i|^2$** if found in i , the new state will be a vector $|t\rangle$ st $t_j = \delta_{j,i}$

Composition: by tensoring on the complex vector space; may exist **entangled** states.

Basic operations

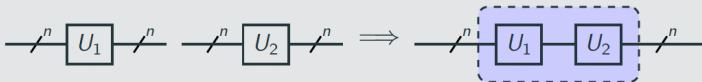
We start with a set of **quantum operations**, e.g.



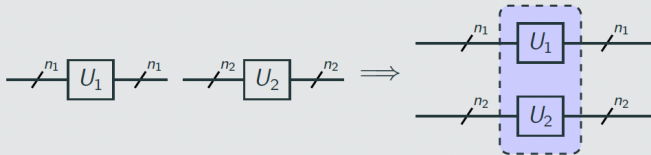
Each operation U_i **manipulates the state** of n_i -qubits received from its left-hand side ... and returns the result on its right-hand side

Composition

Sequential Composition



Parallel Composition



◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡ ▶ ↺ 🔍 ↻

My first quantum algorithm

The Deutsch problem

Decide whether

$$f : 2 \longrightarrow 2$$

is constant or not, with a single evaluation of f ?

- Classically, to determine which case $f(1) = f(0)$ or $f(1) \neq f(0)$ holds requires running f twice
- Resorting to quantum computation, however, it suffices to run f once due to two quantum effects: **superposition** and **interference**

Turning f into a quantum operation

$f : \mathbf{2} \longrightarrow \mathbf{2}$ extends to a linear map $\mathbb{C}^2 \rightarrow \mathbb{C}^2$

... but not necessarily to a **unitary** transformation.

proof

The extended f does not preserve norms: Actually, when f is constant on 0 we obtain $f|0\rangle = |0\rangle$ and $f|1\rangle = |0\rangle$.

Thus,

$$\left| \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \right| = 1$$

However,

$$\left| f \left(\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \right) \right| = \left| \frac{1}{\sqrt{2}}(|0\rangle + |0\rangle) \right| = \left| \frac{2}{\sqrt{2}}|0\rangle \right| = \sqrt{2}$$

Turning f into a quantum operation

Proposed Solution

$$\left[\text{---} \text{---}^2 \boxed{U_f} \text{---}^2 \text{---} \right] = |x\rangle \otimes |y\rangle \mapsto |x\rangle \otimes |y \oplus f(x)\rangle$$



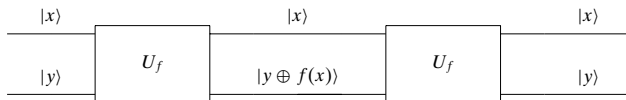
Addition modulo 2

- The **oracle** takes input $|x\rangle|y\rangle$ to $|x\rangle|y \oplus f(x)\rangle$
- Fixing $y = 0$ it encodes f :

$$U_f(|x\rangle \otimes |0\rangle) = |x\rangle \otimes |0 \oplus f(x)\rangle = |x\rangle \otimes |f(x)\rangle$$

Turning f into a quantum operation

- U_f is a **unitary**, i.e. a **reversible** gate

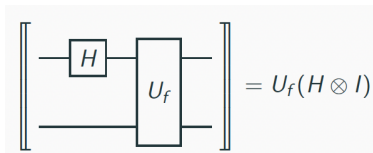


$$|x\rangle|(y \oplus f(x)) \oplus f(x)\rangle = |x\rangle|y \oplus (f(x) \oplus f(x))\rangle = |x\rangle|y \oplus 0\rangle = |x\rangle|y\rangle$$

Exploiting quantum parallelism

Can f be evaluated for $|0\rangle$ and $|1\rangle$ in one step?

Consider the following circuit



$$U_f(H \otimes I)(|0\rangle \otimes |0\rangle)$$

$$= U_f\left(\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |0\rangle\right)$$

{Defn. of H and I }

$$= U_f\left(\frac{1}{\sqrt{2}}(|00\rangle + |10\rangle)\right)$$

{ $\otimes$ distributes over $+$ }

$$= \frac{1}{\sqrt{2}}(|0\rangle|0 \oplus f(0)\rangle + |1\rangle|0 \oplus f(1)\rangle)$$

{Defn. of U_f }

$$= \frac{1}{\sqrt{2}}(|0\rangle|f(0)\rangle + |1\rangle|f(1)\rangle)$$

{ $0 \oplus x = x$ }

$f(0)$ and $f(1)$ in a single run

Are we done?

$$U_f(H \otimes I)(|0\rangle \otimes |0\rangle) = \underbrace{\frac{1}{\sqrt{2}}(|0\rangle|f(0)\rangle + |1\rangle|f(1)\rangle)}_{f(0) \text{ and } f(1) \text{ in a single run}}$$

NO

Although both values have been computed **simultaneously**, only one of them is retrieved upon **measurement** in the computational basis: Actually, 0 or 1 will be retrieved with **identical** probability (why?).

YES

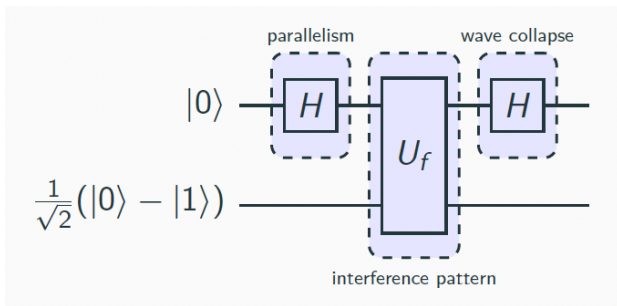
The Deutsch problem is not interested on the concrete values f may take, but on a **global** property of f : whether it is constant or not, technically on the value of

$$f(0) \oplus f(1)$$

Exploiting quantum parallelism and interference

Actually, the **Deutsch algorithm** explores another quantum resource — **interference** — to obtain that **global** information on f

Let us create an **interference pattern** dependent on this property resorting to our **golden pattern**:



Exploiting quantum parallelism and interference

Let us start with a simple, auxiliary computation:

$$\begin{aligned} & U_f (|x\rangle \otimes (|0\rangle - |1\rangle)) \\ &= U_f (|x\rangle|0\rangle - |x\rangle|1\rangle) && \{\otimes \text{ distributes over } + \} \\ &= |x\rangle|0 \oplus f(x)\rangle - |x\rangle|1 \oplus f(x)\rangle && \{\text{Defn. of } f\} \\ &= |x\rangle|f(x)\rangle - |x\rangle|\neg f(x)\rangle && \{0 \oplus x = x, 1 \oplus x = \neg x\} \\ &= |x\rangle \otimes (|f(x)\rangle - |\neg f(x)\rangle) && \{\otimes \text{ distributes over } +\} \\ &= \begin{cases} |x\rangle \otimes (|0\rangle - |1\rangle) & \text{if } f(x) = 0 \\ |x\rangle \otimes (|1\rangle - |0\rangle) & \text{if } f(x) = 1 \end{cases} && \{\text{case distinction}\} \\ &= (-1)^{f(x)} |x\rangle \otimes (|0\rangle - |1\rangle) && \{\text{unfolding the conditional}\} \end{aligned}$$

Exploiting quantum parallelism and interference

$$\begin{aligned} & (H \otimes I) U_f (H \otimes I) (|0\rangle \otimes |-\rangle) \\ &= (H \otimes I) U_f (|+\rangle \otimes |-\rangle) \\ &= \frac{1}{\sqrt{2}} (H \otimes I) U_f (|0\rangle + |1\rangle \otimes |-\rangle) \\ &= \frac{1}{\sqrt{2}} (H \otimes I) (U_f |0\rangle |-\rangle + U_f |1\rangle |-\rangle) \\ &= \frac{1}{\sqrt{2}} (H \otimes I) \left(\frac{1}{\sqrt{2}} U_f (|0\rangle (|0\rangle - |1\rangle)) + \frac{1}{\sqrt{2}} U_f (|1\rangle (|0\rangle - |1\rangle)) \right) \\ &= \frac{1}{\sqrt{2}} (H \otimes I) \left(\frac{1}{\sqrt{2}} (-1)^{f(0)} |0\rangle (|0\rangle - |1\rangle) + \frac{1}{\sqrt{2}} (-1)^{f(1)} |1\rangle (|0\rangle - |1\rangle) \right) \\ &= \frac{1}{\sqrt{2}} (H \otimes I) ((-1)^{f(0)} |0\rangle |-\rangle + (-1)^{f(1)} |1\rangle |-\rangle) \\ &= \begin{cases} (H \otimes I) (\pm 1) |+\rangle |-\rangle & \text{if } f(0) = f(1) \\ (H \otimes I) (\pm 1) |-\rangle |-\rangle & \text{if } f(0) \neq f(1) \end{cases} \\ &= \begin{cases} (\pm 1) |0\rangle |-\rangle & \text{if } f(0) = f(1) \\ (\pm 1) |1\rangle |-\rangle & \text{if } f(0) \neq f(1) \end{cases} \end{aligned}$$

Enough to measure the first qubit: if in state $|0\rangle$, then f is constant.

Lessons learnt

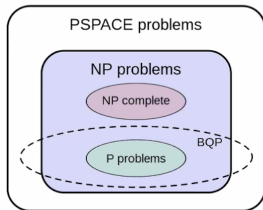
- A typical structure for a quantum algorithm includes three phases:
 1. **State preparation**
(fix initial setting)
 2. **Transformation**
(combination of unitary transformations, typically a variant of our **golden pattern**)
 3. **Measurement**
(projection onto a basis vector associated with a measurement tool)
- This 'toy' algorithm is an illustrative simplification of the first algorithm with **quantum advantage**

presented in literature [Deutsch, 1985]
- All other quantum algorithms crucially rely on similar ideas of quantum interference

Algorithms for quantum advantage

Quantum computers are conjectured to provide **exponential** advantage for **specific** computational problems.

- New **complexity classes** can be defined relevant to quantum computation (theory).
- Algorithmic patterns exclusive to quantum computation make the difference (practice).



(Nielsen & Chuang, 2010)

Algorithms for quantum advantage

The quest

- **Non exponential speedup.** Not relevant for the complexity debate, but shed light on what a quantum computer can do.
Example: Grover's search of an unsorted data base.
- **Exponential speedup relative to an oracle.** By feeding quantum superpositions to an oracle, one can learn what is inside it with an exponential speedup.
Example: Simon's algorithm for finding the period of a unction.
- **Exponential speedup for apparently hard problems**
Example: Shor's factoring algorithm.