Lecture 2:
Introduction to quantum algorithms

Luís Soares Barbosa

Universidade do Minho

HASLab
HIGH-ASSURANCE
SOFTWARE LABORATORY

INL
INTERNATIONAL IBERIAN
NANOTECHNOLOGY
LABORATORY

UNU

Mestrado em Engenharia Física

Universidade do Minho, 2025-26

Physics of information

Information

is encoded in the state of a physical system

Computation

is carried out on an actual physically realizable device

- the study of information and computation cannot ignore the underlying physical processes.
- ... although progress in Computer Science has been made by abstracting from the physical reality
- more precisely: by building more and more abstract models of a sort of reality, i.e. a way of understanding it
- ... until now ...

Physics of information

How physics constrains our ability to use and manipulate information?

- Landauer's principle (1961): information deleting is necessarily a dissipative process.
- Charles Bennett (1973): any computation can be performed in a reversible way, and so with no dissipation.

NAND $\Longrightarrow$ Toffoli

$(x, y) \mapsto \neg(x \wedge y)$

$(x, y, z) \mapsto (x, y, z \oplus (x \wedge y))$
with $z = 1$

Physics of information

Information is physical, and the physical reality is quantum mechanical:

How does quantum theory shed light on the nature of information?

- Quantum dynamics is truly random
- Acquiring information about a physical system disturbs its state (which is related to quantum randomness)
- Noncommuting observables cannot simultaneously have precisely defined values: the uncertainty principle
- Quantum information cannot be copied with perfect fidelity: the no-cloning theorem (Wootters, Zurek, Dieks, 1982)
- Quantum information is encoded in nonlocal correlations between the different parts of a physical system, i.e. the predictions of quantum mechanics cannot be reproduced by any local hidden variable theory (John Bell, 1967)

A model for quantum computation

States

State of n-qubits encoded as a unit vector

$$v \in \underbrace{\mathbb{C}^2 \otimes \cdots \otimes \mathbb{C}^2}_{n \text{ times}} \cong \mathbb{C}^{2^n}$$

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.

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.

A model for quantum computation

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

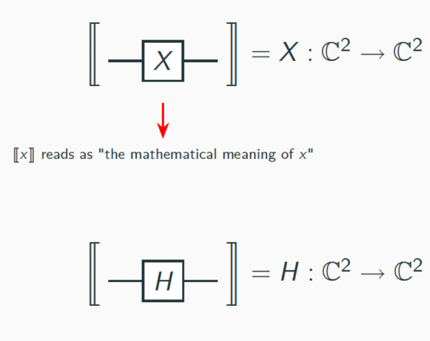
- $M|u\rangle$ (next state)

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: also by a tensor on the complex vector space; may exist entangled states.

Basic operations

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



$$[\![-\boxed{X}-]\!] = X : \mathbb{C}^2 \to \mathbb{C}^2$$

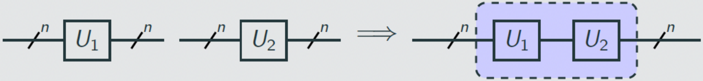
$[\![x]\!]$ reads as "the mathematical meaning of x"

$$[\![-\boxed{H}-]\!] = H : \mathbb{C}^2 \to \mathbb{C}^2$$

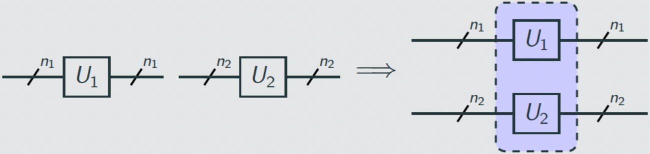
Each operation U_i manipulates the state of n_i-qubits received from its left-hand side . . . and returns the result on its right-hand side

Composition

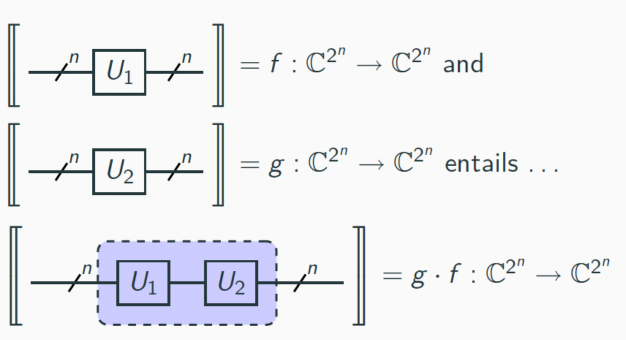
Sequential Composition



Parallel Composition



What does sequential composition mean?

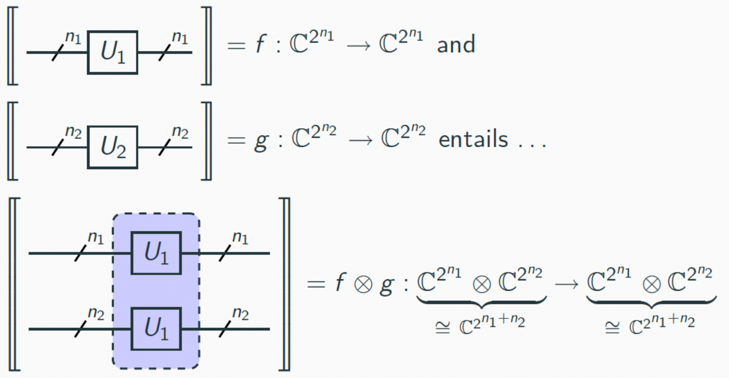


$= f : \mathbb{C}^{2^n} \to \mathbb{C}^{2^n}$ and

$= g : \mathbb{C}^{2^n} \to \mathbb{C}^{2^n}$ entails . . .

$= g \cdot f : \mathbb{C}^{2^n} \to \mathbb{C}^{2^n}$

What does parallel composition mean?



$= f : \mathbb{C}^{2^{n_1}} \to \mathbb{C}^{2^{n_1}}$ and

$= g : \mathbb{C}^{2^{n_2}} \to \mathbb{C}^{2^{n_2}}$ entails . . .

$$= f \otimes g : \underbrace{\mathbb{C}^{2^{n_1}} \otimes \mathbb{C}^{2^{n_2}}}_{\cong\, \mathbb{C}^{2^{n_1+n_2}}} \to \underbrace{\mathbb{C}^{2^{n_1}} \otimes \mathbb{C}^{2^{n_2}}}_{\cong\, \mathbb{C}^{2^{n_1+n_2}}}$$

My first quantum algorithm

The Deutsch problem

Decide whether

$$f : \mathbf{2} \longrightarrow \mathbf{2}$$

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 $\mathbb{C}^2 \to \mathbb{C}^2$

... but not necessarily to a unitary transformation.

proof

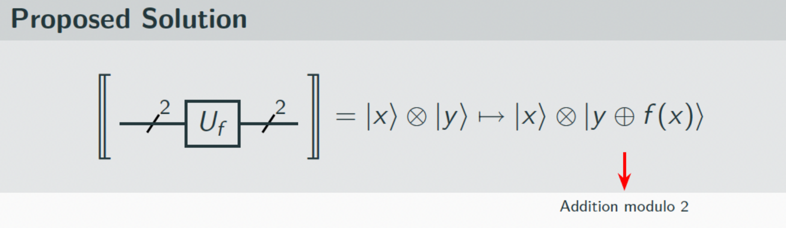
The extended f does not preserve norms: Actually, when f is constant on 0 we obtain $f|0\rangle = |0\rangle$ and $f|1\rangle = |0\rangle$.
Thus,

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

However,

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

Turning f into a quantum operation



Proposed Solution

$$= |x\rangle \otimes |y\rangle \mapsto |x\rangle \otimes |y \oplus f(x)\rangle$$

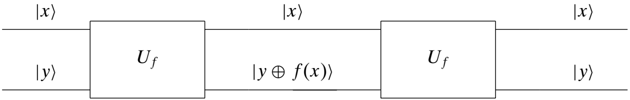
Addition modulo 2

- 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

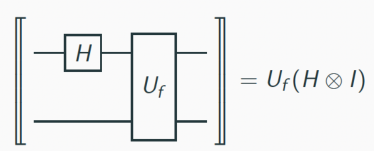


$$|x\rangle|(y \oplus f(x)) \oplus f(x)\rangle \;=\; |x\rangle|y \oplus (f(x) \oplus f(x))\rangle \;=\; |x\rangle|y \oplus 0\rangle \;=\; |x\rangle|y\rangle$$

Exploiting quantum parallelism

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

Consider the following circuit



$$
\begin{aligned}
& U_f(H \otimes I)(|0\rangle \otimes |0\rangle) \\
&= U_f\left(\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |0\rangle\right) && \{\text{Defn. of } H \text{ and } I\} \\
&= U_f\left(\frac{1}{\sqrt{2}}(|00\rangle + |10\rangle)\right) && \{\otimes \text{ distributes over } +\} \\
&= \frac{1}{\sqrt{2}}(|0\rangle|0 \oplus f(0)\rangle + |1\rangle|0 \oplus f(1)\rangle) && \{\text{Defn. of } U_f\} \\
&= \underbrace{\frac{1}{\sqrt{2}}(|0\rangle|f(0)\rangle + |1\rangle|f(1)\rangle)}_{f(0) \text{ and } f(1) \text{ in a single run}} && \{0 \oplus x = x\}
\end{aligned}
$$

Are we done?

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

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

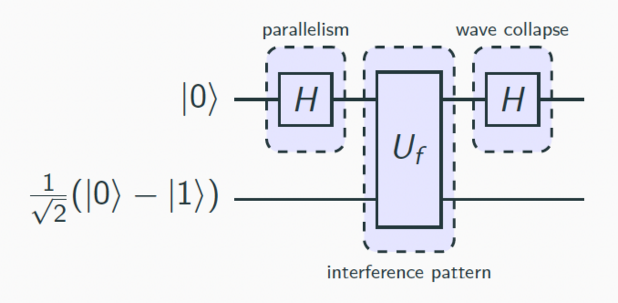
The Deutsch problem is not interested on the concrete values f may take, but on a global property of f: whether it is constant or not, technically on the value of

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

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:

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

leading to

$$U_f\,(|x\rangle \otimes (|0\rangle - |1\rangle)) \;=\; (-1)^{f(x)}|x\rangle \otimes (|0\rangle - |1\rangle)$$

Exploiting quantum parallelism and interference

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

To answer the original problem is now enough to measure the first qubit: if it is 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

Second thoughts

The example illustrates how the golden pattern embodies a basic principle in algorithmic design.

Two notes on

- Function evaluation
- Generating a suitable interference

Boolean function evaluation

Boolean function evaluation is encoded as an oracle:

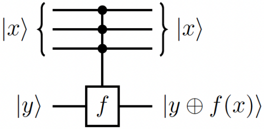
$$|x\rangle|y) \;\mapsto\; |x\rangle|y \oplus f(x)\rangle\rangle$$

which is a special case of a generalised bit-flip (or negation) gate controlled by the function argument:

$$\sum_{x\in\{0,1\}^n} |x\rangle\langle x|\; X^{f(x)}$$

where $X^{f(x)}$ is the identity I (when $f(x) = 0$) or X (when $f(x) = 1$).

Thus, the oracle U_f can be represented as



Boolean function evaluation: Example

Let $f : \{0,1\}^2 \longrightarrow \{0,1\}$ be such that $f(01) = 1$ and evaluates to 0 otherwise.
Oracle $U_f(|x\rangle|y\rangle) = |x\rangle|y \oplus f(x)\rangle$ can be tabulated as

$$\begin{array}{lll@{\qquad}lll}
|00\rangle|0\rangle & \mapsto & |00\rangle|0\rangle & |00\rangle|1\rangle & \mapsto & |00\rangle|1\rangle \\
|01\rangle|0\rangle & \mapsto & |01\rangle|1\rangle & |01\rangle|1\rangle & \mapsto & |01\rangle|0\rangle \\
|10\rangle|0\rangle & \mapsto & |10\rangle|0\rangle & |10\rangle|1\rangle & \mapsto & |10\rangle|1\rangle \\
|11\rangle|0\rangle & \mapsto & |11\rangle|0\rangle & |11\rangle|1\rangle & \mapsto & |11\rangle|1\rangle
\end{array}$$

which corresponds to

$$\sum_{x\in\{0,1\}^2} |x\rangle\langle x|\, X^{f(x)} =$$

$$= |00\rangle\langle 00| \otimes I + |01\rangle\langle 01| \otimes X + |10\rangle\langle 10| \otimes I + |11\rangle\langle 11| \otimes I$$

Boolean function evaluation: Example

Or, in matrix format,

$$U_f = \begin{bmatrix} I & 0 & 0 & 0 \\ 0 & X & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Generating a suitable interference

What is new in quantum evaluation of Boolean functions is the ability to act on a superposition, e.g.

$$\sum_x |x\rangle|0\rangle \;\mapsto\; \sum_x |x\rangle|f(x)\rangle$$

i.e. all results are computed in a single execution

But much more interesting is the effect of starting with $|-\rangle$:

$$\sum_x |x\rangle|-\rangle \;\mapsto\; \sum_x (-1)^{f(x)}|x\rangle|-\rangle$$

which indeed generates the suitable interference

more to follow

Concluding

The meaning of computable remains the same

A classical computer can simulate a quantum computer to arbitrarily good accuracy.

... but the order of complexity may change

However, simulation is computationally hard, i.e. extremely inefficient as the number of qubits increases:

- For 100 qubits the state space would require to store $2^{100} \approx 10^{30}$ complex numbers!
- And what about rotating a vector in a vector space of dimension 10^{30}?

Concluding

In a sense this might not be a decisive argument:

Simulating the evolution of a vector in an exponentially large space can be done locally through a probabilistic classical algorithm in which each qubit has a value at each time step, and each quantum gate can act on the qubits in various possible ways, one of which is selected as determined by a (pseudo)-random number generator.

... After all, the computation provides a means of assigning probabilities to all the possible outcomes of the final measurement...

Concluding

However, Bell's result precludes such a simulation: there is no local probabilistic algorithm that can reproduce the conclusions of quantum mechanics.

In the presence of entanglement, one can access only an exponentially small amount of information by looking at each subsystem separately.

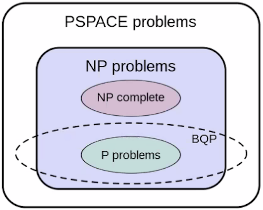
Quantum computing as using quantum reality as a computational resource

Richard Feynman, *Simulating Physics with Computers* (1982)

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.