Lecture 4:
Introduction to quantum algorithms

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

Universidade do Minho

Quantum Data Science
Universidade do Minho
2025-2026

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)

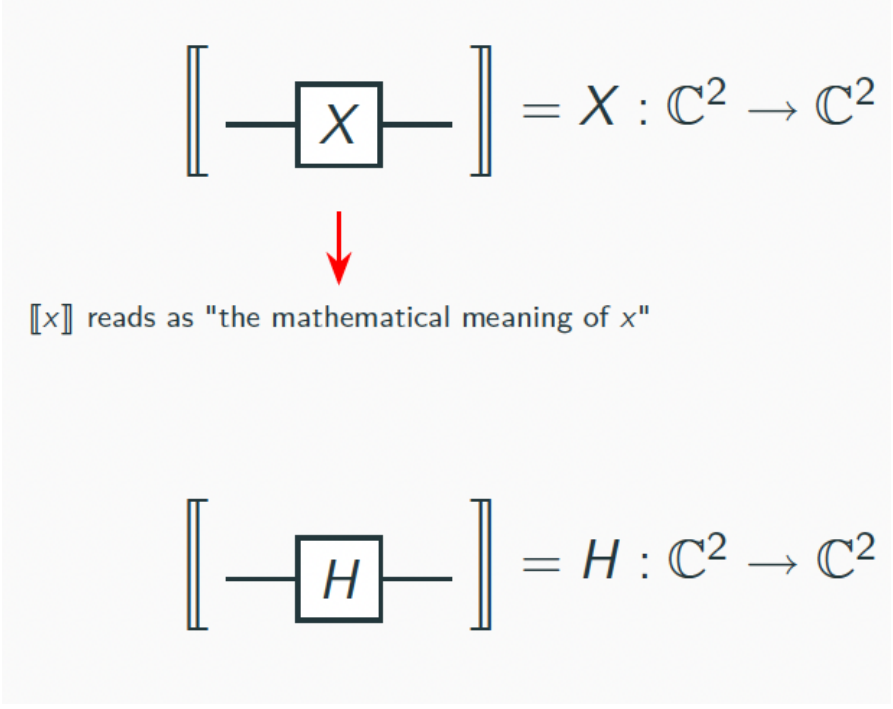
- $M|u\rangle$ (next state)

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

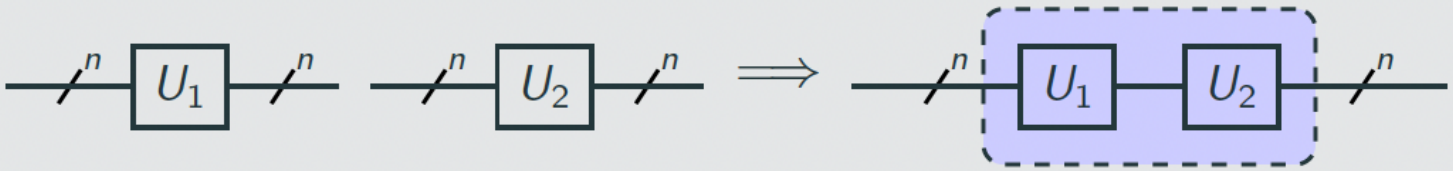
We start with a set of quantum operations, e.g.



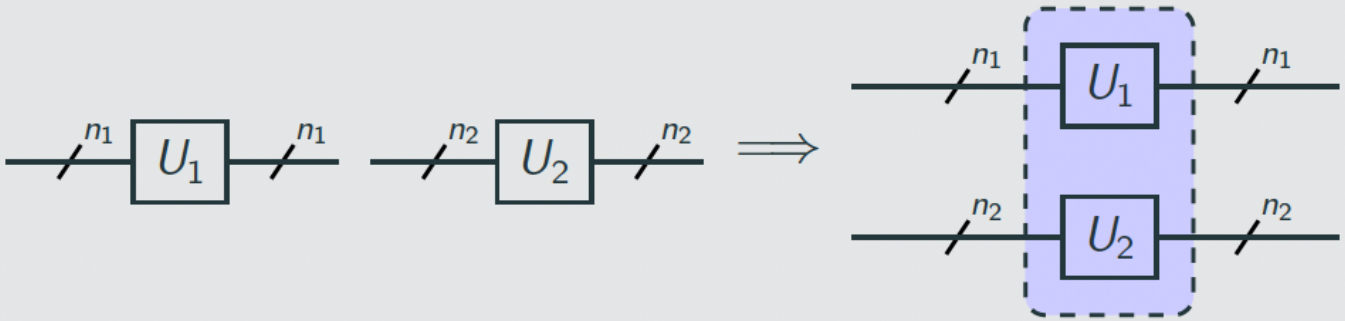
Each operation U_i manipulates the state of n_i-qubits received from its left-hand side . . . and returns the result on its right-hand side

Composition

Sequential Composition



Parallel Composition



What does sequential composition mean?

$$\left[\!\!\left[\; \overset{n}{/} \boxed{U_1} \overset{n}{/} \; \right]\!\!\right] = f : \mathbb{C}^{2^n} \to \mathbb{C}^{2^n} \text{ and}$$

$$\left[\!\!\left[\; \overset{n}{/} \boxed{U_2} \overset{n}{/} \; \right]\!\!\right] = g : \mathbb{C}^{2^n} \to \mathbb{C}^{2^n} \text{ entails } \ldots$$

$$\left[\!\!\left[\; \overset{n}{/} \boxed{U_1}\boxed{U_2} \overset{n}{/} \; \right]\!\!\right] = g \cdot f : \mathbb{C}^{2^n} \to \mathbb{C}^{2^n}$$

What does parallel composition mean?

$$\left[\!\!\left[\; \overset{n_1}{/} \boxed{U_1} \overset{n_1}{/} \; \right]\!\!\right] = f : \mathbb{C}^{2^{n_1}} \to \mathbb{C}^{2^{n_1}} \text{ and}$$

$$\left[\!\!\left[\; \overset{n_2}{/} \boxed{U_2} \overset{n_2}{/} \; \right]\!\!\right] = g : \mathbb{C}^{2^{n_2}} \to \mathbb{C}^{2^{n_2}} \text{ entails } \ldots$$

$$\left[\!\!\left[\begin{array}{l} \overset{n_1}{/} \boxed{U_1} \overset{n_1}{/} \\ \overset{n_2}{/} \boxed{U_1} \overset{n_2}{/} \end{array} \right]\!\!\right] = f \otimes g : \underbrace{\mathbb{C}^{2^{n_1}} \otimes \mathbb{C}^{2^{n_2}}}_{\cong \mathbb{C}^{2^{n_1+n_2}}} \to \underbrace{\mathbb{C}^{2^{n_1}} \otimes \mathbb{C}^{2^{n_2}}}_{\cong \mathbb{C}^{2^{n_1+n_2}}}$$

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,

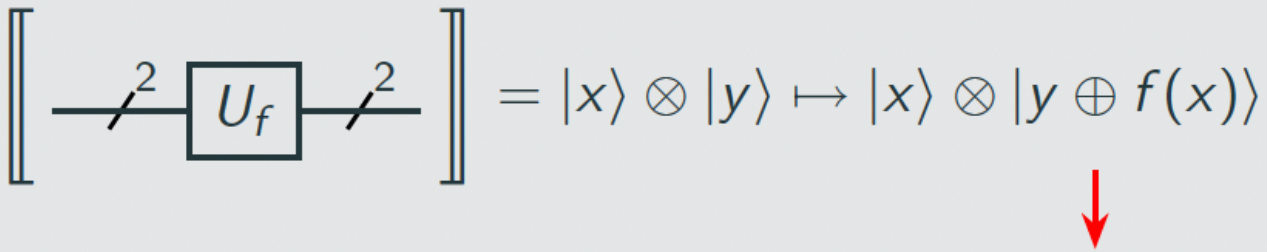
$$\left| \tfrac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \right| = 1$$

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

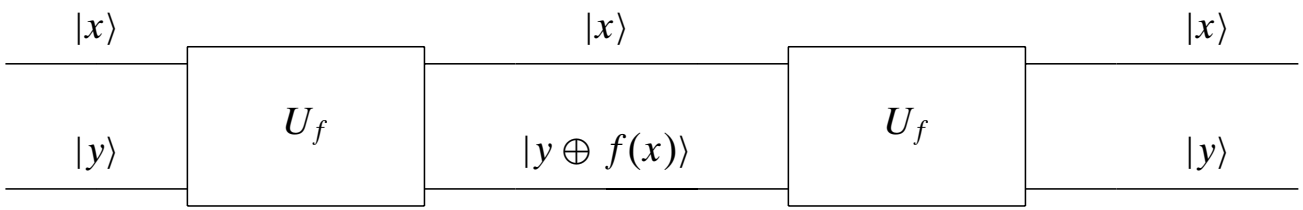


- The oracle takes input $|x\rangle|y\rangle$ to $|x\rangle|y \oplus f(x)\rangle$
- Fixing $y = 0$ it encodes f:

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

Turning f into a quantum operation

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

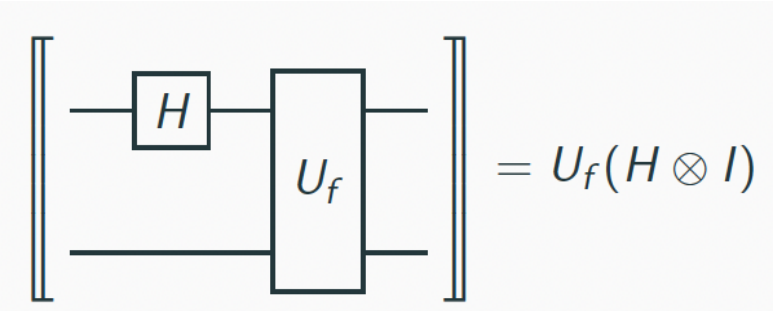


$$|x\rangle|(y \oplus f(x)) \oplus f(x)\rangle = |x\rangle|y \oplus (f(x) \oplus f(x))\rangle = |x\rangle|y \oplus 0\rangle = |x\rangle|y\rangle$$

Exploiting quantum parallelism

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

Consider the following circuit



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

Are we done?

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

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

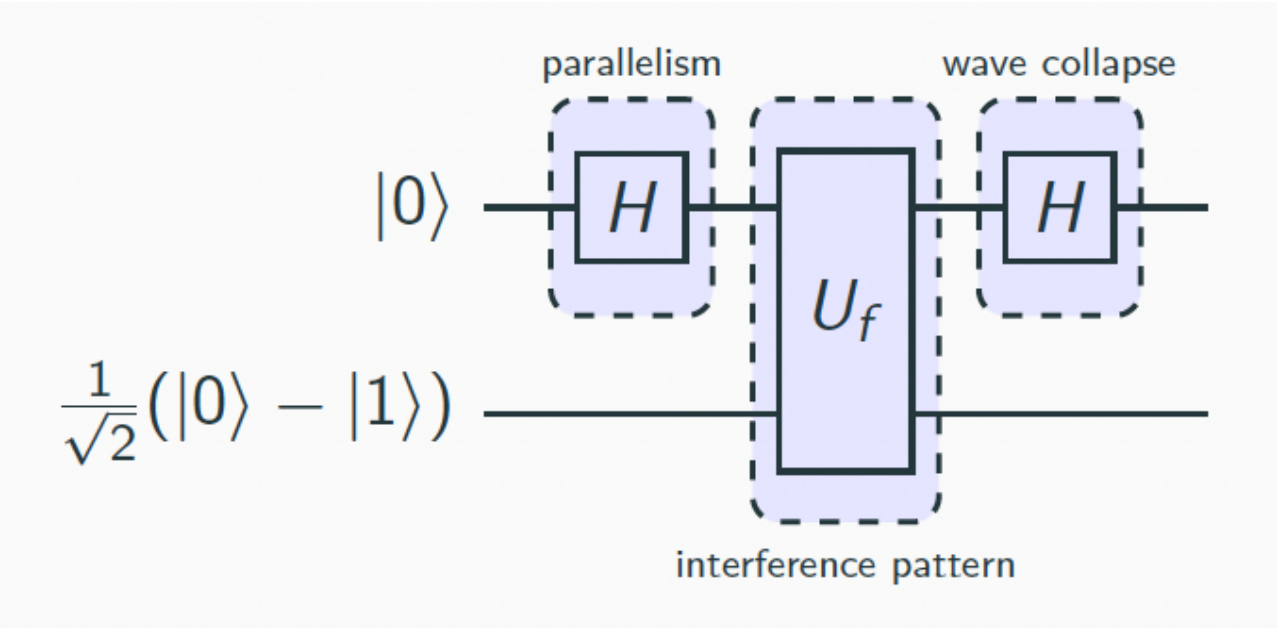
The Deutsch problem is not interested on the concrete values f may take, but on a global property of f: whether it is constant or not, technically on the value of

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

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:

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

leading to

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

Exploiting quantum parallelism and interference

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

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:

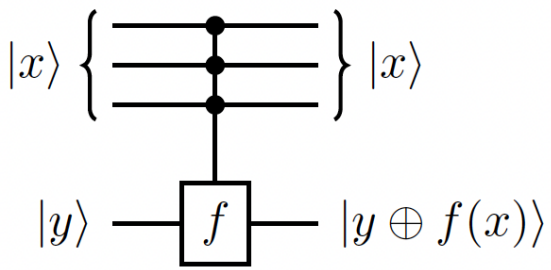
$$|x\rangle|y\rangle \;\mapsto\; |x\rangle|y \oplus f(x)\rangle\rangle$$

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

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

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

The Bernstein-Vazirani algorithm

Let $2^n = \{0,1\}^n = \{0,1,2,\cdots 2^n - 1\}$ be the set of non-negative integers (represented as bit strings up to n bits). Then, consider the following problem:

The problem

Let s be an unknown non-negative integer less than 2^n, encoded as a bit string, and consider a function $f : \{0,1\}^n \to \{0,1\}$ which hides secret s as follows: $f(x) = x \cdot s$, where $\cdot$ is the bitwise product of x and s modulo 2. i.e.

$$x \cdot s \;=\; x_1 s_1 \oplus x_2 s_2 \oplus \cdots \oplus x_n s_n$$

Find s.

Note that juxtaposition abbreviates conjunction, i.e. $x_1 s_1 \;=\; x_1 \wedge s_1$

Setting the stage

Lemma

(1) *For $a, b \in 2$ the equation $(-1)^a(-1)^b = (-1)^{a \oplus b}$ holds.*

Proof sketch

Build a truth table for each case and compare the corresponding contents.

Lemma

(2) *For any three binary strings $x, a, b \in 2^n$ the equation $(x \cdot a) \oplus (x \cdot b) = x \cdot (a \oplus b)$ holds.*

Proof sketch

Follows from the fact that for any three bits $a, b, c \in 2$ the equation $(a \wedge b) \oplus (a \wedge c) = a \wedge (b \oplus c)$ holds.

Setting the stage

Lemma

(3) *For any element $|b\rangle$ in the computational basis of $\mathbb{C}^2$,*

$$H|b\rangle = \frac{1}{\sqrt{2}} \sum_{z \in 2} (-1)^{b \wedge z} |z\rangle$$

Proof sketch

Build a truth table and compare the corresponding contents.

Theorem

(1) *For any element $|b\rangle$ in the computational basis of $\mathbb{C}^{2^n}$,*

$$H^{\otimes n}|b\rangle = \frac{1}{\sqrt{2^n}} \sum_{z \in 2^n} (-1)^{b \cdot z} |z\rangle$$

Proof sketch

Follows by induction on the size of n.

The Bernstein-Vazirani algorithm

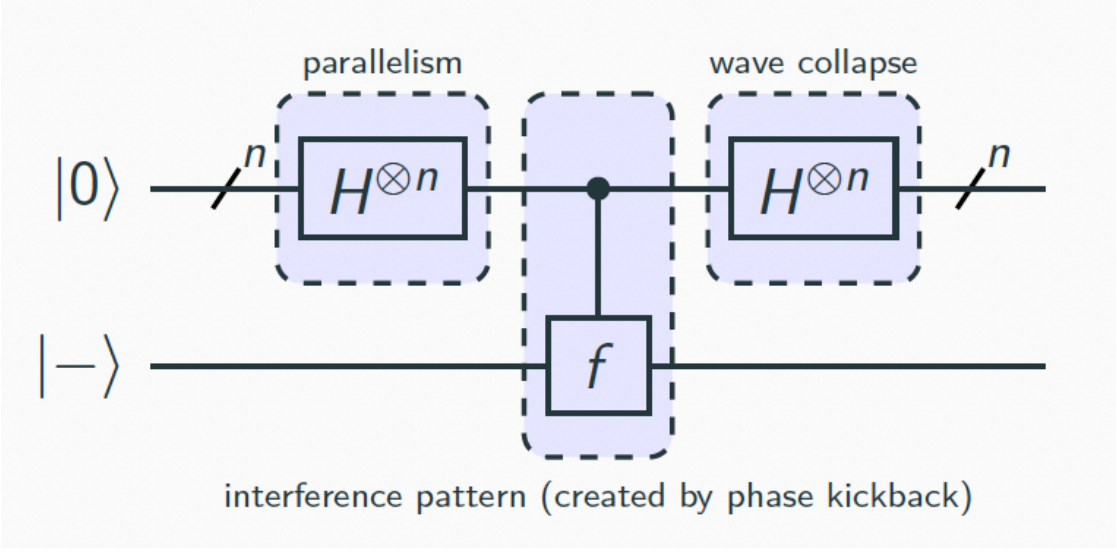
How many times f has to be called to determine s?

- Classically, we run f n-times by computing

$$f(1\ldots0) = (s_1 \wedge 1) \oplus \cdots \oplus (s_n \wedge 0) = s_1$$
$$\vdots$$
$$f(0\ldots1) = (s_1 \wedge 0) \oplus \cdots \oplus (s_n \wedge 1) = s_n$$

- With a quantum algorithm, we may discover s by running f only once

The circuit


parallelism
wave collapse
$|0\rangle$
$|-\rangle$
n
$H^{\otimes n}$
f
$H^{\otimes n}$
n
interference pattern (created by phase kickback)

The computation

$$
\begin{aligned}
& H^{\otimes n}|0\rangle|-\rangle \\
&= \tfrac{1}{\sqrt{2^n}} \textstyle\sum_{z\in 2^n} |z\rangle|-\rangle && \{\text{Theorem (1)}\} \\
&\overset{U_f}{\mapsto} \quad \tfrac{1}{\sqrt{2^n}} \textstyle\sum_{z\in 2^n} (-1)^{f(z)}|z\rangle|-\rangle && \{\text{Definition}\} \\
&\overset{H^{\otimes n}\otimes I}{\mapsto} \quad \tfrac{1}{2^n} \textstyle\sum_{z\in 2^n} (-1)^{f(z)} \left(\sum_{z'\in 2^n} (-1)^{z\cdot z'}|z'\rangle \right)|-\rangle && \{\text{Theorem (1)}\} \\
&= \tfrac{1}{2^n} \textstyle\sum_{z\in 2^n} \sum_{z'\in 2^n} (-1)^{(z\cdot s)\oplus(z\cdot z')}|z'\rangle|-\rangle && \{\text{Lemma (1)}\} \\
&= \tfrac{1}{2^n} \textstyle\sum_{z\in 2^n} \sum_{z'\in 2^n} (-1)^{z\cdot(s\oplus z')}|z'\rangle|-\rangle && \{\text{Lemma (2)}\} \\
&= |s\rangle|-\rangle && \{\text{Why?}\}
\end{aligned}
$$

Why?

$$\cdots = \frac{1}{2^n} \sum_{z \in 2^n} \sum_{z' \in 2^n} (-1)^{z \cdot (s \oplus z')} |z'\rangle |-\rangle = \cdots$$

For each z, $\frac{1}{2^n} \sum_{z=0}^{2^n-1} (-1)^{z \cdot (s \oplus z')}$ is 1 iff $(s \oplus z') = 0$, which happens only if $s = z'$ In all other cases $\frac{1}{2^n} \sum_{z=0}^{2^n-1} (-1)^{z \cdot (s \oplus z')}$ is 0.

The reason is easy to guess:

- for $s \oplus z' = 0$, $\frac{1}{2^n} \sum_{z=0}^{2^n-1} (-1)^{z \cdot (s \oplus z')} = \frac{1}{2^n} \sum_{z=0}^{2^n-1} 1 = 1$.
- for $s \oplus z' \neq 0$, as z spans all numbers from 0 to $2^n - 1$, half of the 2^n factors in the sum will be -1 and the other half 1, thus summing up to 0.

Thus, the only non zero amplitude is the one associated to s.

Why?

Alternatively, consider the probability of measuring s at the end of the computation:

$$\begin{aligned}
&\left|\tfrac{1}{2^n}\textstyle\sum_{z\in 2^n}(-1)^{z\cdot(s\oplus s)}\right|^2 \\
&= \left|\tfrac{1}{2^n}\textstyle\sum_{z\in 2^n}(-1)^{z\cdot 0}\right|^2 \\
&= \left|\tfrac{1}{2^n}\textstyle\sum_{z\in 2^n} 1\right|^2 \\
&= \left|\tfrac{2^n}{2^n}\right|^2 \\
&= 1
\end{aligned}$$

This means that somehow all values yielding wrong answers were completely cancelled.