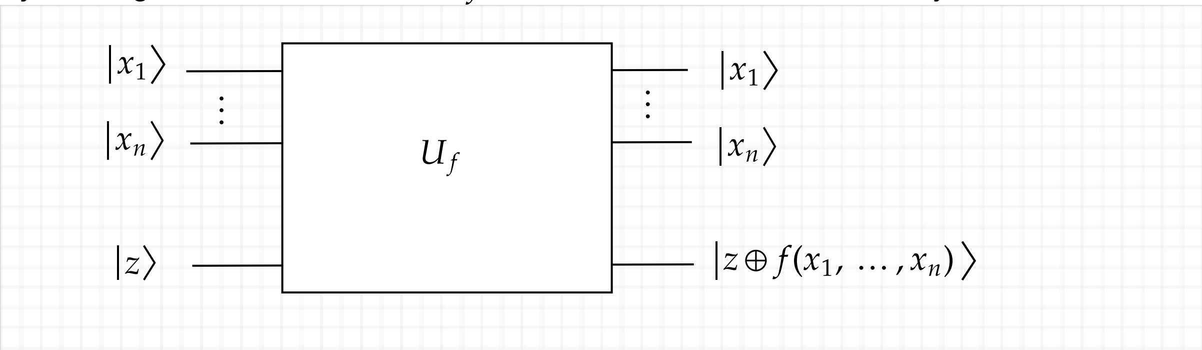
Oracles, measures and Deutsch-Josza

Computational power: simulating boolean circuits

Ojective : given a boolean function $f\ :\ \mathcal{B}ool^n \to \mathcal{B}ool$, realize a unitary circuit



The wires x_1,... x_n correspond to the input variables of the function j, and the z is the output register.

Note : It is indeed a unitary.
- U_f sends a basis vector to a basis vector.
- The corresponding function is reversible:
  $\overline{f}\ :\ (\vec{x}, z)\ \mapsto (\vec{x},\ z \oplus f(\vec{x}))$

Indeed,
$\overline{f}(\overline{f}(x,z))\ \ =\ \ \overline{f}(x,\ z \oplus f(x))$
$\ =\ \ (x,\ z \oplus f(x) \oplus f(x))\ \ =\ \ (x,z)$
(she is its own inverse)

In general, such a box is called an **oracle:** it captures the (classical) structure of the problem instance. For instance, it can correspond to an arithmetic operation, or the neighboring relation for a graph, etc.

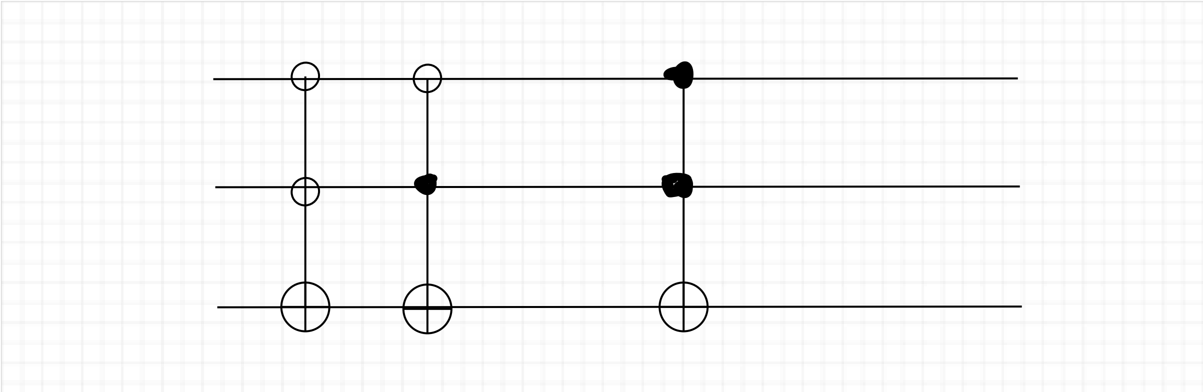
How to build such a U_f ?

$\to$ THE WHOLE POINT is to get a circuit with a reasonnable size....

It depends how f is provided... If given as a truth table, the description of f is exponential compared to the input size. For instance, if f takes 2 values and is defined as

input	00	01	10	11
f	1	1	0	1

one can build U_f as follows



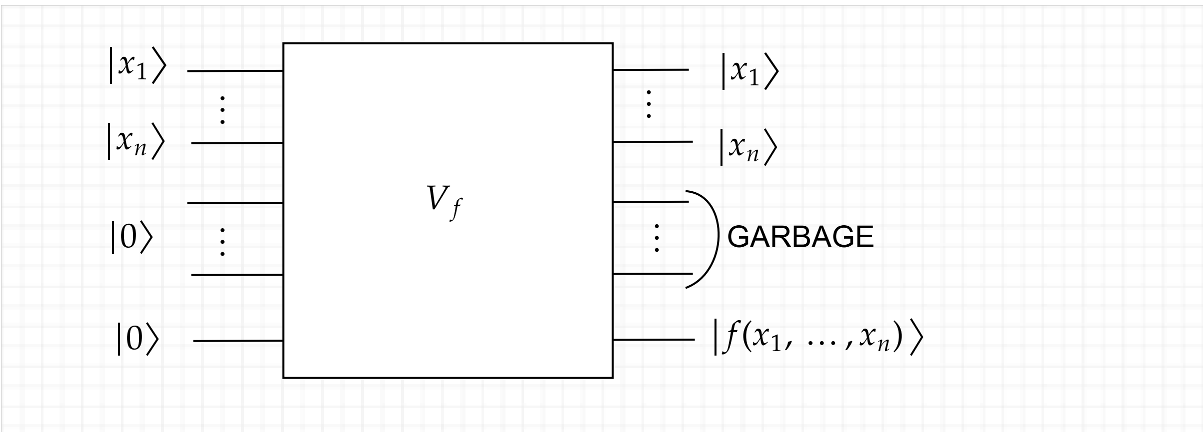
This is not optimal in general.

If f is given as a boolean formula, one can then build a polynomial-size circuit compared to the size of the formula.

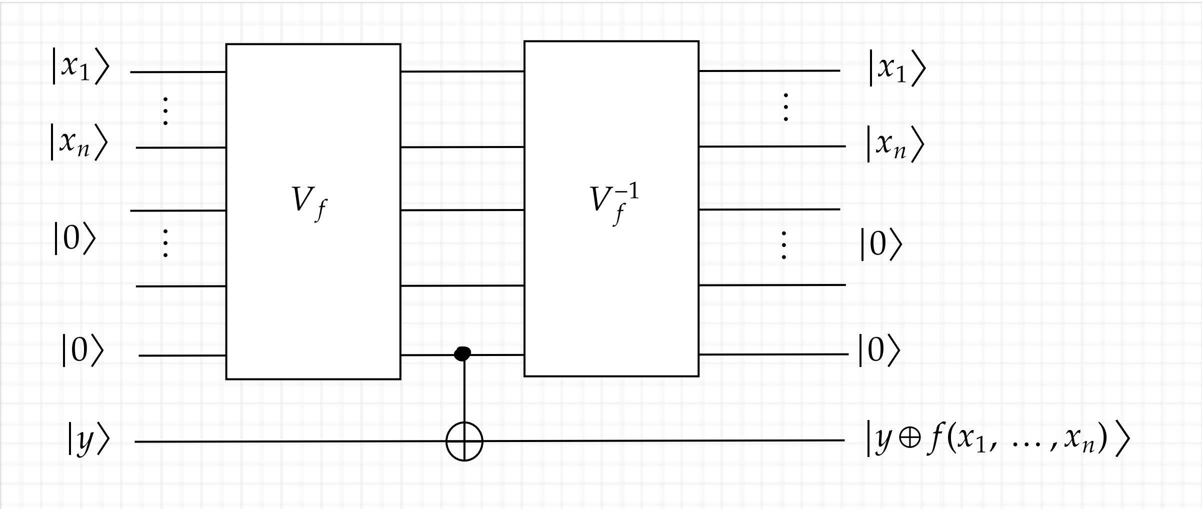
The function f is typically built from
 - conjunctions → implementable with Toffolis
 - negations → implementable with NOTs and CNOTs
 - composition → circuit composition

The procedure is in two steps.

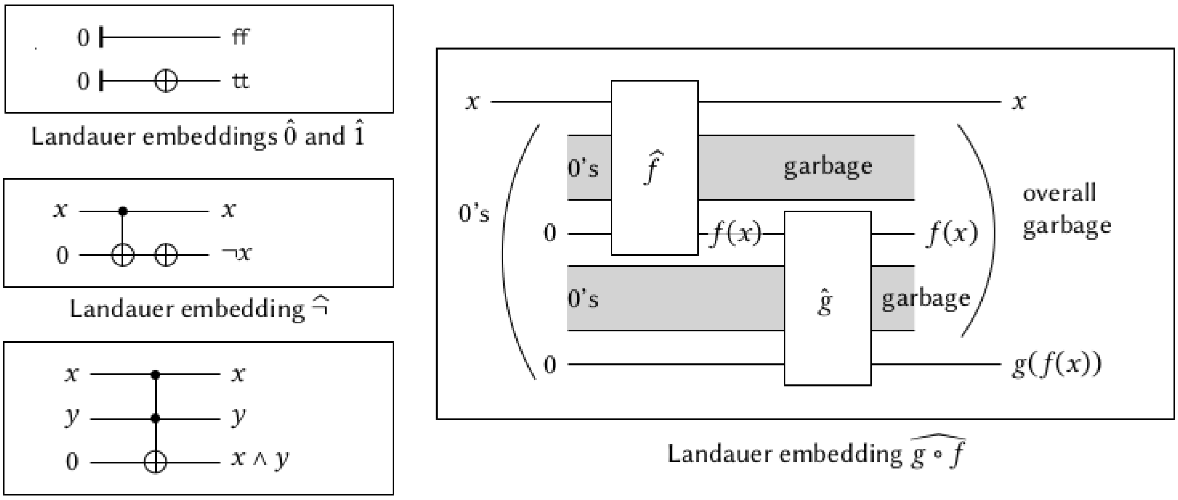
First, let us (compositionally) build V_f as follows:



Then we can build U_f as



V_f is built as follows.



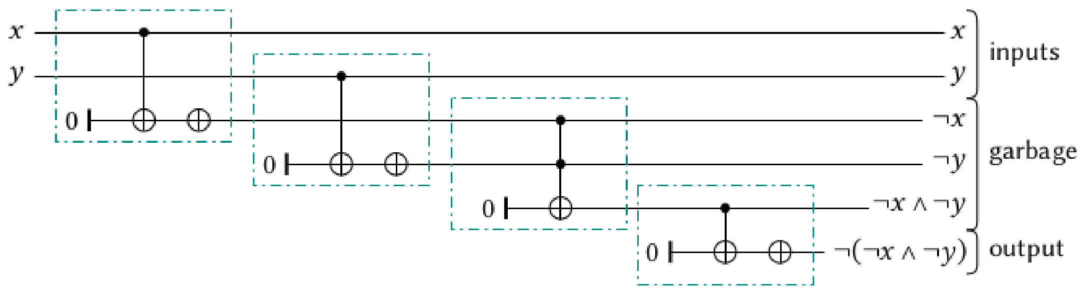
Landauer embeddings $\hat{0}$ and $\hat{1}$

Landauer embedding $\widehat{\neg}$

Landauer embedding $\hat{\wedge}$

Landauer embedding $\widehat{g \circ f}$

Exercice: with the map $f\,:\,(x,y) \mapsto \neg(\neg x \wedge \neg y)$



Coming back to U_f

$f : \mathcal{B}ool^n \to \mathcal{B}ool$ a boolean function

$\overline{f} : (\vec{x}, z) \mapsto (\vec{x}, z \oplus f(\vec{x}))$ a reversible function

We saw how to realize a unitary map U_f computing $\overline{f}$ on quantum registers

This operation U_f is built from V_f acting on 3 registers:

- input register $|\vec{x}\rangle$ (with n qubits)
- "garbage" registers with auxiliary wires initialized at 0
- register to store the output $|z\rangle$

So V_f is an operator acting on $\mathcal{H}^{\otimes n} \otimes \mathcal{H}^{\otimes garbage} \otimes \mathcal{H}$ (last one is output register)

This is not exactly U_f which should act on $\mathcal{H}^{\otimes n} \otimes \mathcal{H}$

U_f is built as

$$\mathcal{H}^{\otimes n} \otimes \mathcal{H}$$
$$\xrightarrow{embedding} \mathcal{H}^{\otimes n} \otimes \mathcal{H}^{\otimes garbage} \otimes \mathcal{H}$$
$$\xrightarrow{V_f} \mathcal{H}^{\otimes n} \otimes \mathcal{H}^{\otimes garbage} \otimes \mathcal{H}$$
$$\xrightarrow{CNOT} \mathcal{H}^{\otimes n} \otimes \mathcal{H}^{\otimes garbage} \otimes \mathcal{H}$$
$$\xrightarrow{V_f^{-1}} \mathcal{H}^{\otimes n} \otimes \mathcal{H}^{\otimes garbage} \otimes \mathcal{H}$$
$$\xrightarrow{\text{"}magic\text{"}} \mathcal{H}^{\otimes n} \otimes \mathcal{H}$$

Why am I allowed to delete this "garbage" register?

Idea : The inner action of U_f can be regarded as an opertaion on
$\mathcal{H}^{\otimes n} \otimes \mathcal{H}^{\otimes garbage} \otimes \mathcal{H}$
preserving the subspace
$\mathcal{H}^{\otimes n} \otimes |0\rangle^{\otimes garbage} \otimes \mathcal{H}$

So globally, we have a permutation of all the chains of bits under the form
$x_1 ... x_n 00000000z$
So also a permutation of the chains of bits of the form
$x_1 ... x_n z$
...
And so it is unitary.

Another way to see it is to write the matrix of V_f when seen as an operator on

$\mathcal{H}^{\otimes garbage} \otimes (\mathcal{H}^{\otimes n} \otimes \mathcal{H})$. The inner operation of U_f can be written blockwise as an action on

- $|0\rangle^{\otimes garbage} \otimes (\mathcal{H}^{\otimes n} \otimes \mathcal{H})$ $\rightarrow$ some operation A
- the rest $\rightarrow$ Some operation B

It has the shape

$$\begin{pmatrix} A & 0 \\ 0 & B \end{pmatrix}$$

(because the $|0\rangle^{\otimes garbage}$ register is sent back to $|0\rangle^{\otimes garbage}$)
As overall it is a unitary, each block is a unitary. "Dropping" the garbage register yields A.

The operation V_f does not in general maintain the $|0\rangle^{\otimes garbage}$ register in its original form.
So V_f is in general of the form

$$\begin{pmatrix} A & B \\ C & D \end{pmatrix}$$

and there is no reason for A to be a unitary...

In general, droping a register makes us leave the realm of unitary maps !

What happens when we "delete" the garbage register then ?

$\rightarrow$ a **measure** is performed

Measure

This is the ONLY way to get back classical data out of quantum data.

Measuring $\alpha \cdot |0\rangle + \beta \cdot |1\rangle$ we obtain

- with prob. $|\alpha|^2$ the value "0" and the qubit is now in state $|0\rangle$
- with prob. $|\beta|^2$ the value "1" and the qubit is now in state $|1\rangle$

The qubit state has been probabilistically **projected** on one basis vector

$\rightarrow$ Measuring $|0\rangle$ returns "0" with probability 1....

As vectors are normalized, the sum of probabilities is indeed equal to 1.

What about when we have several qubits ?

With 2 qubits: $\alpha|00\rangle + \beta|01\rangle + \gamma|10\rangle + \delta|11\rangle$
In this case, if we measure the 1st qubit, we project
- either on the subspace $|0\rangle \otimes \mathcal{H}$ where the first qubit is $|0\rangle$, spanned by $|00\rangle, |01\rangle$

- or on the subspace $|1\rangle \otimes \mathcal{H}$ where the first qubit is $|1\rangle$, spanned by $|10\rangle$, $|11\rangle$

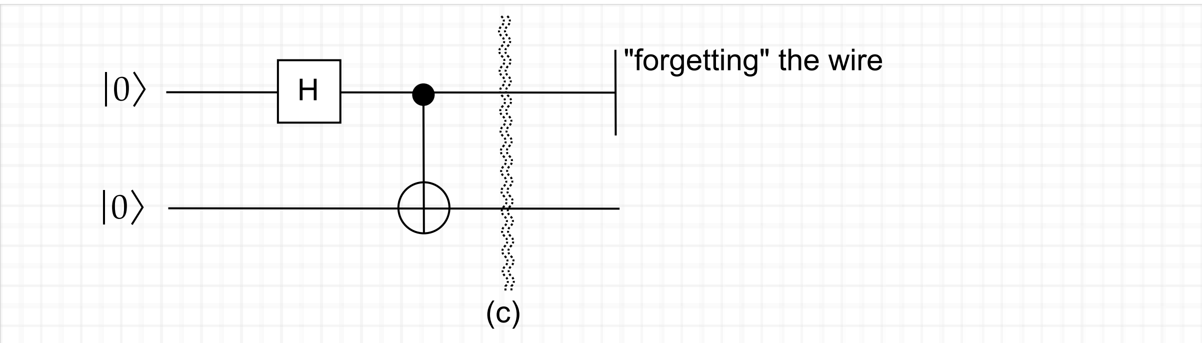
We get
- value "0" and a state of the form $\alpha|00\rangle + \beta|01\rangle$ (modulo renormalisation) with probability $|\alpha|^2 + |\beta|^2$
- value "1" and a state of the form $\gamma|10\rangle + \delta|11\rangle$ (modulo renormalisation) with probability $|\gamma|^2 + |\delta|^2$

Measuring the second qubit, we end up with the 4 possibilities:
- value "00" with the state now at $|00\rangle$ with probability $|\alpha|^2$
- value "01" with the state now at $|01\rangle$ with probability $|\beta|^2$
- value "10" with the state now at $|10\rangle$ with probability $|\gamma|^2$
- value "11" with the state now at $|11\rangle$ with probability $|\delta|^2$

Note : one can measure qubits in an arbitrary order, this does not change the final result.

Unitarity, auxiliary wires and measures

When we "forget" a wire, there is an implicit measure
For instance :



At (c) : state is $|\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$

Right after forgetting the wire, there is only one qubit left, the second one → first wire got measured without keeping track of the result of the measurement (we speak of a **partial trace**).

In our setting, the result of a partial trace is an equal probabilistic distribution of $|0\rangle$ et $|1\rangle$, each with probability 1/2.

We projected the state $|\psi\rangle$ on : either $|0\rangle \otimes \mathcal{H}$, either $|1\rangle \otimes \mathcal{H}$
We splitted $|\psi\rangle$ in $\alpha|\psi_0\rangle + \beta|\psi_1\rangle$ with $|\psi_0\rangle \in |0\rangle \otimes \mathcal{H}$ and $|\psi_1\rangle \in |1\rangle \otimes \mathcal{H}$

Here, α and β are both equal to $\frac{1}{\sqrt{2}}$

So beware, this is not the state $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$!!

Corollary: be careful with auxiliary wires ! in general, before deleting them, we need them to be non-entangled (separated) with the rest of the system.

Other thing to be aware of: in general, forgetting a wire (or measuring) "breaks" unitarity.

Indeed, we go from linear operators on vector spaces to more general operations on probability distributions.

Beware

The two probability distributions

A = $\frac{1}{2}\left\{ |0\rangle \right\} + \frac{1}{2}\left\{ |1\rangle \right\}$

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

are not the same !

Indeed : apply Hadamard followed with a measurement.

On A : In half of the cases, $|0\rangle \xrightarrow{Hadamard} \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ followed with a measurement: got true and false with prob. 1/2

In the other cases, $|1\rangle \xrightarrow{Hadamard} \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$ followed with a measurement: got true and false with prob. 1/2

→ global behavior is an unbiased coin

On B : In the only case, $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \xrightarrow{Hadamard} |0\rangle$ then measure : with prob 1 we get false.

One can distinguish between A and B : they are not the same state.

However, with measure one can build unitary maps.... if we play well. The first example is the gate U_f we saw (since forgetting a register ≡ measuring it), but this is a bit cheating since the result of the measurement is not used.

Some simple quantum circuits

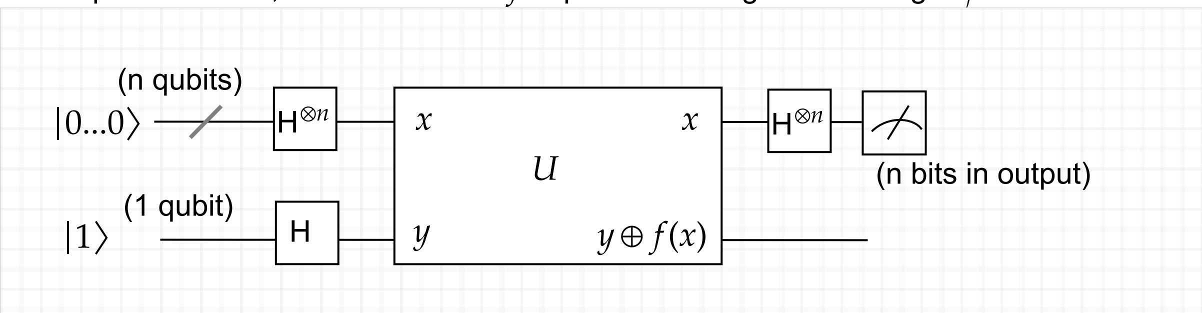
Deutsch-Josza algorithm

Suppose that f : bool^n $\longrightarrow$ bool is either constant, either balanced (its "quality").
(balanced means : $f^{-1}(1)$ and f^{-1}(0) have the same size)
(said otherwise: size($\{i \mid f(i) = 0\}$) = size($\{i \mid f(i) = 1\}$))

Question : how to decide on the quality of f ?
$\rightarrow$ we only consider the quantity fo call to the **oracle**, i.e. the number of calls to f seen as a blackbox (we do not care how it was implemented)

In the classical case we would need at least $2^{n-1} + 1$ calls to f.

In the quantum case, we assume that f is provided using its encoding U_f and we do



f is constant if $|0...0\rangle$ is measured, and balanced otherwise.

Note : The box U_f is the **oracle** of the algorithm