Introduction to Quantum Information Processing

Lecture 2

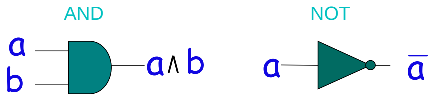
Michele Mosca

Overview

- "Classical" Logic Gates
- Reversible Logic
- Quantum Gates
- A taste of quantum algorithms: Deutsch algorithm

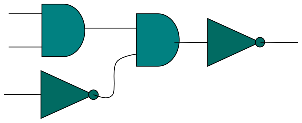
"Classical" Logic Gates (3.1.2)

- A gate is a function from m bits to n bits, for some fixed numbers m and n



“Classical” Logic Gates

- We “glue” gates together to make “circuits” (or “arrays of gates”) which compute Boolean functions



Universal Set of Logic Gates

- A set **B** of gates is universal if, for any Boolean function F, there is a circuit with gates in **B** that computes F
- E.g. B = { NOT } is not universal
- E.g B = { AND } is not universal
- E.g. B = {NOT, AND } is universal

Universal Set of Logic Gates

- A circuit designed with one finite set ***A*** of gates can be efficiently translated into a circuit using gates from a universal set **B.**
- How? Note that since **B** is universal, every gate in ***A*** can be realised by a circuit composed of gates from **B**. So we simply replace each gate *G* in ***A*** with an appropriate circuit of gates from **B**.

“Classical” Logic Gates

- If all physical processes are unitary (and thus reversible), a complete description of a physical process implementing the AND gate should be reversible.
- However the AND gate is not logically reversible.
- Therefore, the (non-reversible) AND gate “throws away” or “erases” information that would make it reversible.

“Classical” Logic Gates

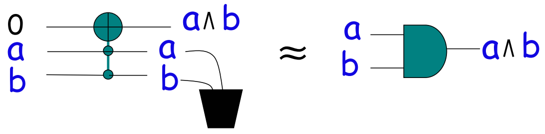
- Landauer’s Principle (3.2.5): To erase a single bit of information dissipates at least kT log(2) amount of energy into the environment
- It was thought that dissipation of energy implied fundamental limits on real computation

“Classical” Logic Gates

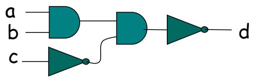
- However Bennett showed that any computation can be made reversible and therefore doesn’t in principle require energy dissipation
- Method: Replace each irreversible gate with a reversible generalization

Irreversible gates from reversible ones

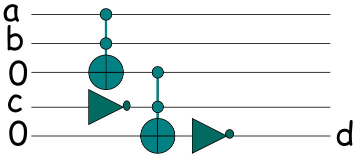
- Note that irreversible gates are really just reversible gates where we hardwire some inputs and throw away some outputs



Making reversible circuits



- Replace irreversible gates with their reversible counterparts



Making reversible circuits

- One problem is that there will be junk left in the extra bits we don't uncompute
- Bennett showed how to uncompute the junk

$$|x\rangle|0\rangle|0\rangle|0\rangle$$

$$\xrightarrow{\text{compute } f(x)} |x\rangle|f(x)\rangle|\text{junk}(x)\rangle|0\rangle$$

$$\xrightarrow{\text{copy } f(x)} |x\rangle|f(x)\rangle|\text{junk}(x)\rangle|f(x)\rangle$$

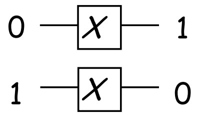
$$\xrightarrow{\text{uncompute } f(x)} |x\rangle|0\rangle|0\rangle|f(x)\rangle$$

Making reversible circuits

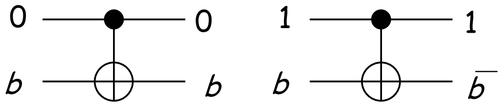
- An irreversible circuit with space S and depth (or "time") T can thus be simulated by a reversible circuit with space in O(S+T) and time O(T)
- Bennett also showed how to implement a reversible version with time $O(T^{1+\varepsilon})$ and space O(S log(T)) or time O(T) and space $O(ST^{\varepsilon})$.

New gates/notation

"X"-gate or NOT-gate



"controlled-NOT" gate



Probabilistic computing

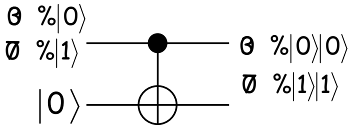
- Suppose we have two bits, corresponding to two distinguishable systems, A and B.
- Suppose we flip a fair coin to establish the value of the bit A.
- We can describe the state of bit A as $\left(50\ \%|0\rangle, 50\ \%|1\rangle\right)$ or simply $(.5, 5)$
- In general, the state of any probabilistic bit can be of the form (a,b) where $0 \le a,b \le 1, a+b=1$

Probabilistic computing

- Note that the NOT or X gate corresponds to multiplying the probability vector by the matrix $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$

Probabilistic computing

- Suppose the state of system (A,B) is $(.3, 7),(1,0)$



- Bits A and B become “correlated”; we cannot describe them independently

Probabilistic computing

- We could describe the four state system (A,B) with one vector $(.3,0,0,7)$
  - The state $(.3, 7),(1,0)$ would correspond to the vector $(.3,0, 7,0)$
  - The controlled-NOT corresponds to multiplying the 4-tuple by $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$

Some tensor product facts

$$\begin{bmatrix} a \\ b \end{bmatrix} \otimes \begin{bmatrix} c \\ d \end{bmatrix} = \begin{bmatrix} a\begin{bmatrix} c \\ d \end{bmatrix} \\ b\begin{bmatrix} c \\ d \end{bmatrix} \end{bmatrix} = \begin{bmatrix} ac \\ ad \\ b \\ bd \end{bmatrix}$$

Some tensor product facts

$$\begin{bmatrix} a_1 & a_2 \\ a_2 & a_2 \end{bmatrix} \otimes \begin{bmatrix} b_1 & b_2 \\ b_2 & b_2 \end{bmatrix} = \begin{bmatrix} a_1\begin{bmatrix} b_1 & b_2 \\ b_2 & b_2 \end{bmatrix} & a_2\begin{bmatrix} b_1 & b_2 \\ b_2 & b_2 \end{bmatrix} \\ a_2\begin{bmatrix} b_1 & b_2 \\ b_2 & b_2 \end{bmatrix} & a_2\begin{bmatrix} b_1 & b_2 \\ b_2 & b_2 \end{bmatrix} \end{bmatrix}$$

$$= \begin{bmatrix} a_1 b_1 & a_1 b_2 & a_2 b_1 & a_2 b_2 \\ a_1 b_2 & a_1 b_2 & a_2 b_2 & a_2 b_2 \\ a_2 b_1 & a_2 b_2 & a_2 b_1 & a_2 b_2 \\ a_2 b_2 & a_2 b_2 & a_2 b_2 & a_2 b_2 \end{bmatrix}$$

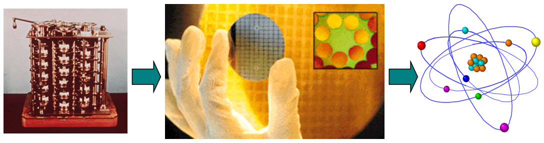
Some tensor product facts

$$\left(\begin{bmatrix} a_1 & a_2 \\ a_2 & a_2 \end{bmatrix}\begin{bmatrix} v_1 \\ v_2 \end{bmatrix}\right) \otimes \left(\begin{bmatrix} b_1 & b_2 \\ b_2 & b_2 \end{bmatrix}\begin{bmatrix} w_1 \\ w_2 \end{bmatrix}\right) =$$

$$= \left(\begin{bmatrix} a_1 & a_2 \\ a_2 & a_2 \end{bmatrix} \otimes \begin{bmatrix} b_1 & b_2 \\ b_2 & b_2 \end{bmatrix}\right)\left(\begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \otimes \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}\right)$$

$$(Av) \otimes (Bw) = (A \otimes B)(v \otimes w)$$

Information and Physics



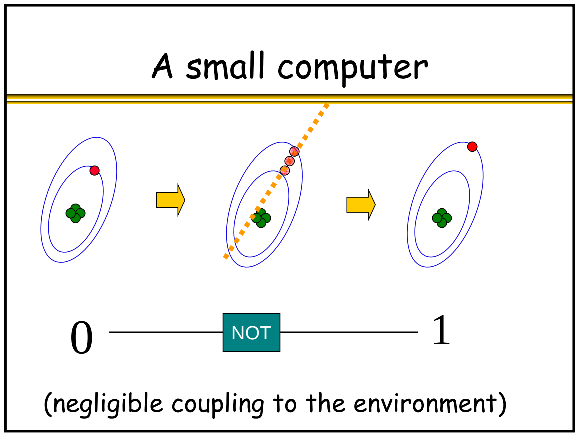
- Information is always stored in a physical medium and manipulated by a physical process.
- Any meaningful theory of information processing must refer (at least implicitly) to a realistic physical theory.

Quantum Mechanics and Information Processing

- Since physics is quantum mechanical, we need to recast the theory of information processing in a quantum mechanical framework.

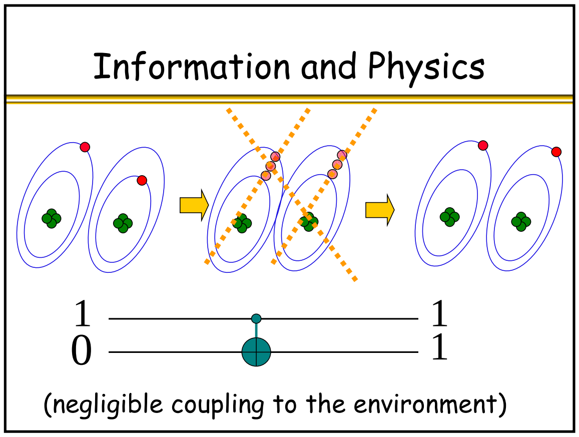
Implications

- Any physical medium capable of representing 0 and 1 is in principle capable of storing any linear combination $\alpha_0|0\rangle + \alpha_1|1\rangle$
- How does this affect computational complexity?
- How does this affect communication complexity?
- How does this affect information security?
- Would you believe a quantum proof?



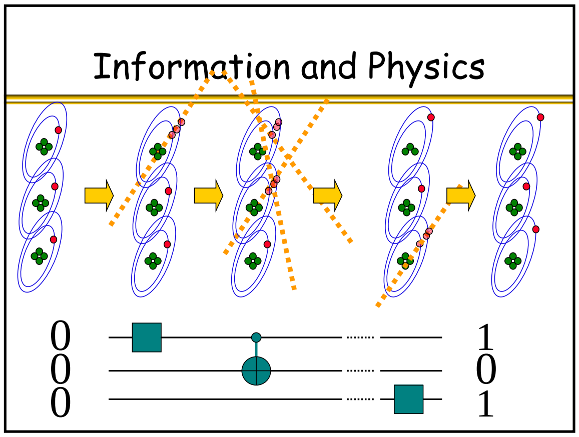
A small computer

(negligible coupling to the environment)

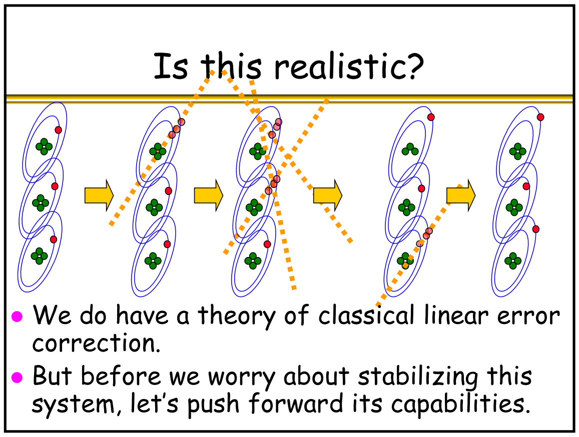


Information and Physics

(negligible coupling to the environment)

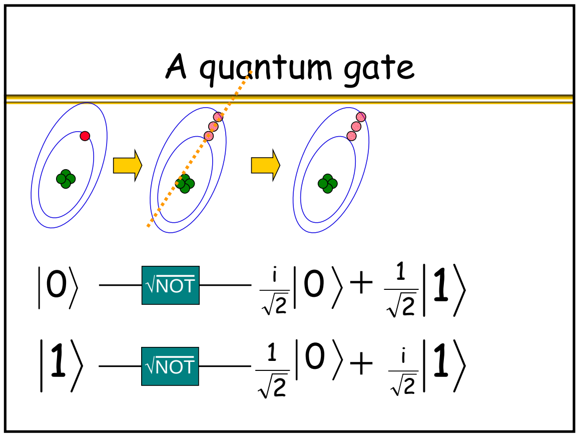


Information and Physics



Is this realistic?

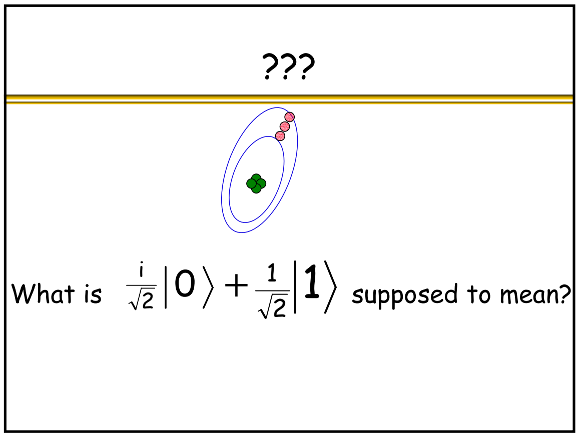
- We do have a theory of classical linear error correction.
- But before we worry about stabilizing this system, let's push forward its capabilities.



A quantum gate

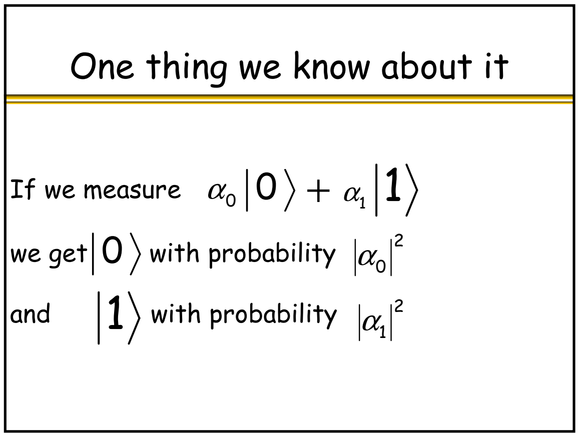
$$|0\rangle \;\text{—}\boxed{\sqrt{\text{NOT}}}\text{—}\; \frac{i}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$$

$$|1\rangle \;\text{—}\boxed{\sqrt{\text{NOT}}}\text{—}\; \frac{1}{\sqrt{2}}|0\rangle + \frac{i}{\sqrt{2}}|1\rangle$$



???

What is $\frac{i}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$ supposed to mean?



One thing we know about it

If we measure $\alpha_0 |0\rangle + \alpha_1 |1\rangle$

we get $|0\rangle$ with probability $|\alpha_0|^2$

and $|1\rangle$ with probability $|\alpha_1|^2$