PH 5840 Lecture-16.

Note Title 9/17/2015

Note: On the definition of complexity classes.

(a) $\{C_n\} \in P \Rightarrow |C_n| \le \text{poly}(n)$

$\uparrow$

A uniform circuit family of Boolean functions.

(b) $L \in P \Rightarrow \exists$ a Turing machine M such that

$\uparrow$ M runs for time $t_n \le \text{poly}(n)$

Language $\forall$ input-strings x.

* Turing Machine: An abstract idealization of a machine that runs a "program" (algorithm)
  - Assumed to have a finite set of states
  - starts in 'q_s'
  - If computation finishes, it ends up in state 'q_h'.
  - o/p is written on a "tape" (sequence of 0's & 1's)

* Church-Turing hypothesis: The class of functions computable by a Turing Machine is equivalent to the class of functions computable by an algorithm.

Reversible computing & Landauer's Principle

* NOT gate : reversible logic gate
* AND, NAND etc : irreversible gates

$\Downarrow$

Information erasure when the gate operates

$\Downarrow$

Energy dissipation

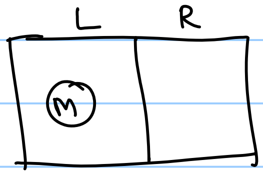
(a) Landauer's principle: If a computing device erases a single bit of information, amount of energy dissipated (to the environment) $\geq k_B T \ln 2$ $\quad (k_B = 1.38 \times 10^{-23} \, m^2 kg \, s^{-2} K^{-1})$ Boltzmann constant

$T \rightarrow$ (temp of environment)

In terms of entropy: If a computing device erases a single bit of information, entropy of the environment ↑eases by $k_B \ln 2$.

Explanation :- Erasure $\Rightarrow$ Shrinking of phase space

For eg,



1 bit $\Leftrightarrow$ whether molecule (M) is in L or R

erasure $\Rightarrow$ Move (M) to R (or L) independent of where it started out.

$\Delta S_{gas} = -k_B \ln 2$

$\Delta S_{env} = k_B \ln 2$

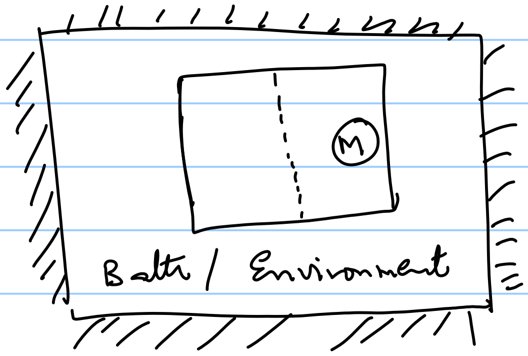
This requires "work": Remove the partition, Then, compress the single-molecule "gas" with a piston until it is in R.

#

A heat-flow from box to environment

Assuming it's an isothermal process,

Work done $= -k_B T \Delta S = k_B T \ln 2$



Box is in eqm with a thermal bath at temperature T.

References:

- Rolf Landauer (1961): Irreversibility & heat generation in the computing process, IBM J Res. Dev. 5, 183 (1961)
- Charles Bennett (1982) - The thermodynamics of computation - a review Int. J. of Theor. Phys. 21, 905 (1982).
  (1987) - Demons, Engines and the second law - Sci. Am 295, 108 (1987)

(b) Reversible classical gates:-

$$f: \{0,1\}^n \longrightarrow \{0,1\}^n$$

→ Permutation of n-bit strings ⇒ $(2^n)!$ such functions

- Recall: total # of such fns $= (2^{2^n})^n = (2^n)^{2^n}$

- Stirling's approximation : $(2^n)! = \left(\frac{2^n}{e}\right)^{2^n}$

  $(1/e^{2n}) \downarrow$

- very small fraction of Boolean functions are reversible

* Universality with reversible gates?

(i) Mapping between reversible $\longleftrightarrow$ irreversible computations.

Example: Given, $f : \{0,1\}^n \to \{0,1\}$

Define, $\tilde{f} : \{0,1\}^{n+1} \to \{0,1\}^{n+1}$:-

$$\tilde{f}(x, y) = (x, y \oplus f(x))$$

$\downarrow$

(flips if $f(x) = 1$)

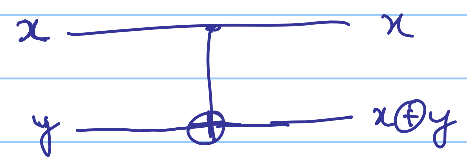
Note, $\tilde{f}$ is invertible!

- To evaluate $f(x)$, set $y = 0$ & read out the last bit of $\tilde{f}(x,y)$.

(ii) 2-bit reversible gate : Modified XOR $\equiv$ CNOT

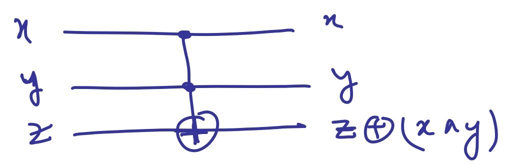
$f(x, y) = (x, x \oplus y)$

($\Lambda(X)$: controlled-X operation)



(iii) 3-bit reversible gate: But the example above indicates that to implement the basic set of universal gates (2-bit gates), we need 3-bit reversible gates.

(a) Toffoli gate:- $T(x, y, z) = (x, y, z \oplus x \wedge y)$



$(\Lambda^2(X))$

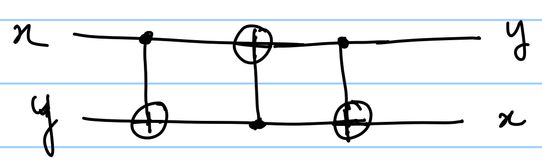
* **Toffoli gate** is a **universal logic gate**!

- implements NOT for z if $x = y = 1$
- implements AND for x, y if $z = 0$
- $\left(x \vee y = \overline{(\bar{x} \wedge \bar{y})}\right)$

(b) **Fredkin gate**: SWAPs first and second bit if third bit is set to 1

x	y	z		x	y	z
0	0	0		0	0	0
0	0	1		0	0	1
0	1	0		0	1	0
0	1	1	→	1	0	1
1	0	0		1	0	0
1	0	1	→	0	1	1
1	1	0		1	1	0
1	1	1		1	1	1

* **SWAP** can be implemented by a series of CNOTs:-



$(x, y) \rightarrow (x, x \oplus y) \rightarrow (y, x \oplus y) \rightarrow (y, x)$

* **Fredkin gate** is also a **universal logic gate**.

⇒ Can implement any Boolean fn. (irreversible) using reversible logic gates. However,

- this requires ancilla bits
- every computation gives out some "junk" bits which clog memory
- At some point, we do need to erase and hence Landauer's principle applies!

- Trade-off between "time" and "space"

* QUANTUM GATES:
- universality
- complexity class BQP
- Algorithms (Grover, Shor)