6.841 Lecture 02

Note Title

Today

- Diagonalization : Power + Problems
- $NTIME(o(n^2)) \subsetneq NTIME(n^2)$
- $P \neq NP \Rightarrow \exists L \in NP - (NP\text{-complete}) - P$

  NP - Intermediate.
- Relativization & Implications.

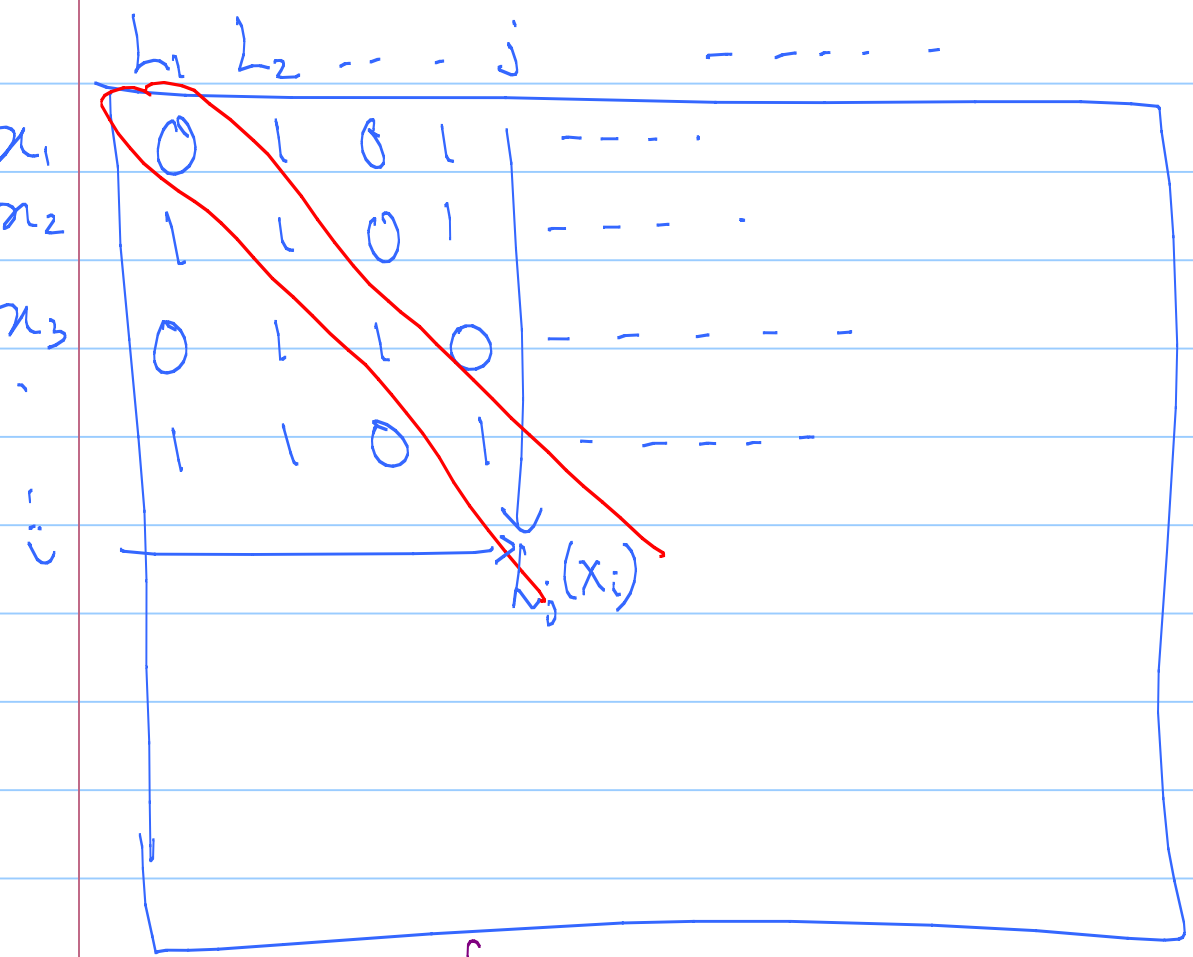
Diagonalization Review

Say want $L \neq \{L_1, L_2, \ldots L_i, \ldots\}$

Notation: $L(x) = 1$ if $x \in L$

$= 0$ o.w.

Construct infinite matrix



$\bar{L} = \overline{\text{Diagonal}} = \{1, 0, 0, 0, \ldots\}$

To ensure L has low complexity must be able to simulate computations for

$L_1, L_2, \ldots L_i, \ldots$

(all of them with one machine)

& must be able to complement answer.

Diagonalization Summary

1. ENUMERATION
2. SIMULATION
3. COMPLEMENTATION

Easy to see $\text{TIME}(n^2) \subsetneq \text{TIME}(n^{10})$ ...

but what about $\text{NTIME}(n^2)$ vs. $\text{NTIME}(n^{10})$?

Can't complement?

Theorem: $NTIME(n^2) \subsetneq NTIME(\omega(n^2))$

[Problem in PSET].

(Theorem above proved by Cook;
Proof from Fortnow's Survey on Diagonalization
Also see van Melkebeek's paper....)

Moving On: Understanding NP

- Would like to show $NP \neq P$ but have failed.
- Almost all problems we've seen are NP-Complete or in P. Why?

- Is this a theorem ?

- Natural counterexamples : Factoring,
  (at the time) LP,
  Graph Isomorphism.

(Just a handful !)

LADNER'S THEOREM : If $NP \neq P$ then there

Exist $L \in NP$

$L \notin P$

$L \notin NP\text{-Complete}$

Notation : M^O = Machine M with access to oracle O (subroutine)

e.g. M^{SAT} = M with alg. for SAT etc.

Proof of LADNER'S THEOREM

Main Idea:

Let $M_1^O, M_2^O, \ldots M_i^O, \ldots$

be enumeration of polytime reductions (algorithms with access to oracle for O);

Will make sure $L \in NP$, — (0)

But $L \neq M_1^\phi, M_2^\phi, \ldots M_i^\phi$ — (1)

ϕ = empty language $\in P$

& $SAT \neq M_1^L, M_2^L, \ldots M_i^L$ .... — (2)

SAT = NP-complete.

(0), (1), (2) $\Rightarrow$ Theorem

Construction of L in stages

j^{th} Stage has parameter i:

j.1 Will try to make sure

$$L \neq M_j^{\phi} : \text{ How?}$$

Input X; unless we find small x s.t.

$$L(x) \neq M_j^{\phi}(x) \quad \text{will let}$$

$$L(X) = SAT(X);$$

j.2 Will try to make sure

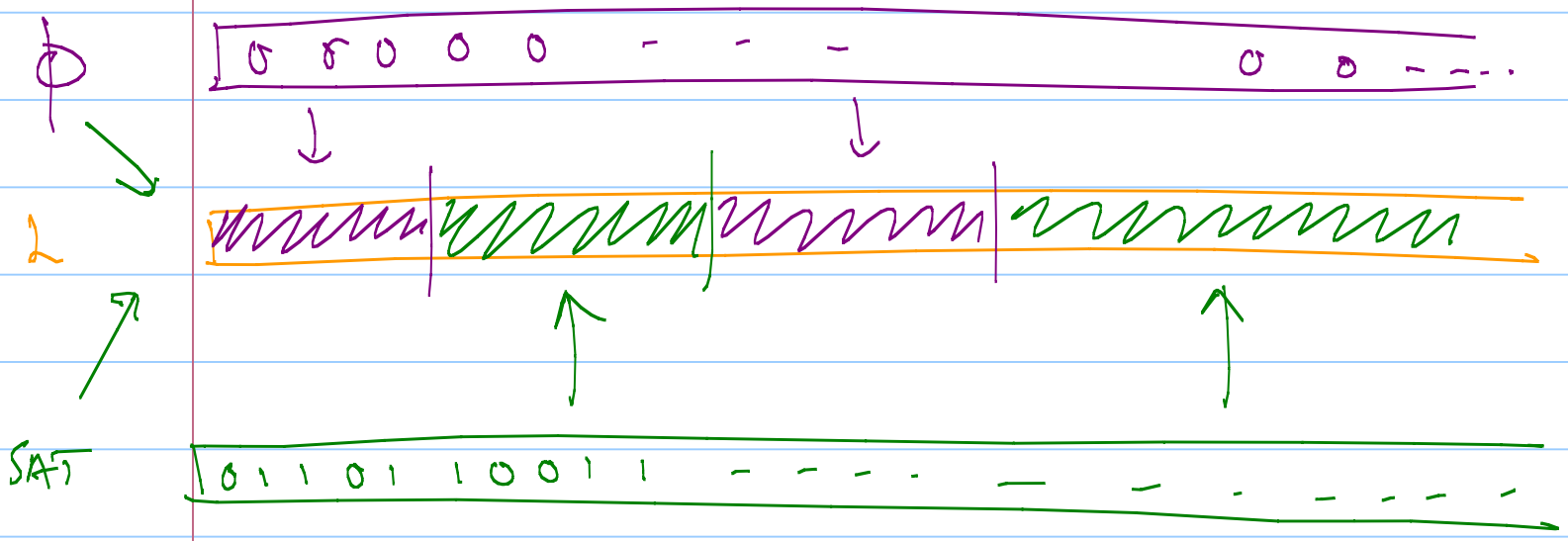
$$SAT \neq M_j^{L} : \text{ How?}$$

Input X: unless we find small x s.t.

$$SAT(x) \neq M_j^{L}(x) \quad \text{will let}$$

$$L(X) = 0;$$

Pictorially



No explicit implementation !

So why is $L \notin P$?

Can only happen if we get in stage $j \cdot 1$ (or $j \cdot 2$).

But if so then $\Rightarrow$ $\forall$ sufficiently large x

- have $L(x) = SAT(x)$; $\leftarrow$ NP

But also $L(x) = M_j^{\Phi}(x)$ $\leftarrow$ P !

Converting to formal proof = Good Exercise (on PS1);

So .... Can Diagonalization Prove NP ≠ P?

Correct Answer : Don't know !

Functional Answer : [Baker Gill Solovay]

Not directly ....

Feature of diagonalization ... relativizes.

i.e. if Diagonalization proves that

$C = \{L(m_1), L(m_2) \ldots L(m_i), \ldots\}$

does not contain $L(M)$

then if also "proves" that for every O

$$C^O = \{ L(M_1^O), L(M_2^O), \ldots L(M_i^O) \ldots \}$$

does not contain $L(M^O)$

M_i^O = M_i with oracle access to oracle O.

As stated above doesn't make sense.

To make sense should define $M_i \triangleq M_i^{\phi}$ (trivial oracle.)

& $M = M^{\phi}$

Assertion 1: if Diagonalization proves $NP \neq P$ then for every O it proves $NP^O \neq P^O$

(By fiat)

Proposition 1: $NP^{TQBF} = P^{TQBF} = PSPACE$

$(\exists\, O \text{ s.t. } NP^O = P^O)$

Proposition 2: $\exists\, O$ s.t. $NP^O \neq P^O$

Proof: Construct O by Diagonalization!

$L^O = \{x \mid$ if $\exists y \quad |y| = |x| \ \& \ O(y) = 1\}$.

$L^O \in NP^O \quad \forall O$.

Need O s.t. $L^O \notin P^O$

To ensure $L^O \neq M_j^O$ on inputs of length $\geq i$

consider queries that $M_j(0^i)$ makes to O; for $O(x)$ $|x| < i$ answer based on whatever was decided.

for $|x| \geq i$ say $\underline{O(x) = 0}$;

Can only ask $p(i)$ such queries; if $p(i) < 2^i$ still have other y $|y| = i$; fix their value to negate the answer $M_i^O(0^i)$.

Poetic Justice

Diagonalization proves its own futility!