CSCE 750
11/30/2023

Proof: (1) Let f be a p-reduction from Π_1 to Π_2. $\forall x$ instance of Π_1, $f(x)$ is an instance of Π_2, and

$$\Pi_1(x) = \text{"yes"} \iff \Pi_2(f(x)) = \text{"yes"}$$

[equiv. $\Pi_1(x) = \text{"no"} \iff \Pi_2(f(x)) = \text{"no"}$.]

Assume $\Pi_2 \in P$, so there is ~~an~~ a ptime A_2 ~~defs~~ deciding Π_2. WTS there is a ptime algo A_1 deciding Π_1:

A_1: On input x, instance of Π_1:
1) Compute $y = f(x)$
2) Run A_2 on input y and return A_2's answer.

Correct? ✓ because of the "$\iff$" condition above.

P-time? Suppose f runs in time $O(n^k)$ some $k \geq 1$
Suppose A_2 runs in time $O(n^\ell)$ some $\ell \geq 1$.

Runtime of A_1: $\underbrace{O(n^k)}_{\text{time for } f} + \underbrace{O(|y|^\ell)}_{\text{time for } A_2}$

2

But $|y| = O(n^k)$ [f has no time to output anything longer]

Runtime of $A_1 = O\big(\underbrace{n^k + (n^k)^\ell}\big) = O(n^{k\ell})$

$\therefore$ A_1 is p-time.

$\therefore$ $\Pi_1 \in P$

$\boxed{//}$ ~~of (1)~~)

(2) f be as before, but now assume $\Pi_2 \in NP$ via a ptime verifier V_2.

Build a p-time verifier V_1 for Π_1:

V_1: On input x instance of Π_1 and string y:

1) Let $z := f(x)$

2) Run V_2 on input z and y and ~~answer~~ accept or reject same as V_2.

Correct ✓ Ptime b/c 1) f is ptime,

2) $|z|$ is $Poly(|x|)$

3) V_2 runs in time $Poly(|z|)$

[similar to part (1)]. $\boxed{//}$

Meaning: $\Pi_1 \leq_p \Pi_2$ means Π_1 is no harder than Π_2

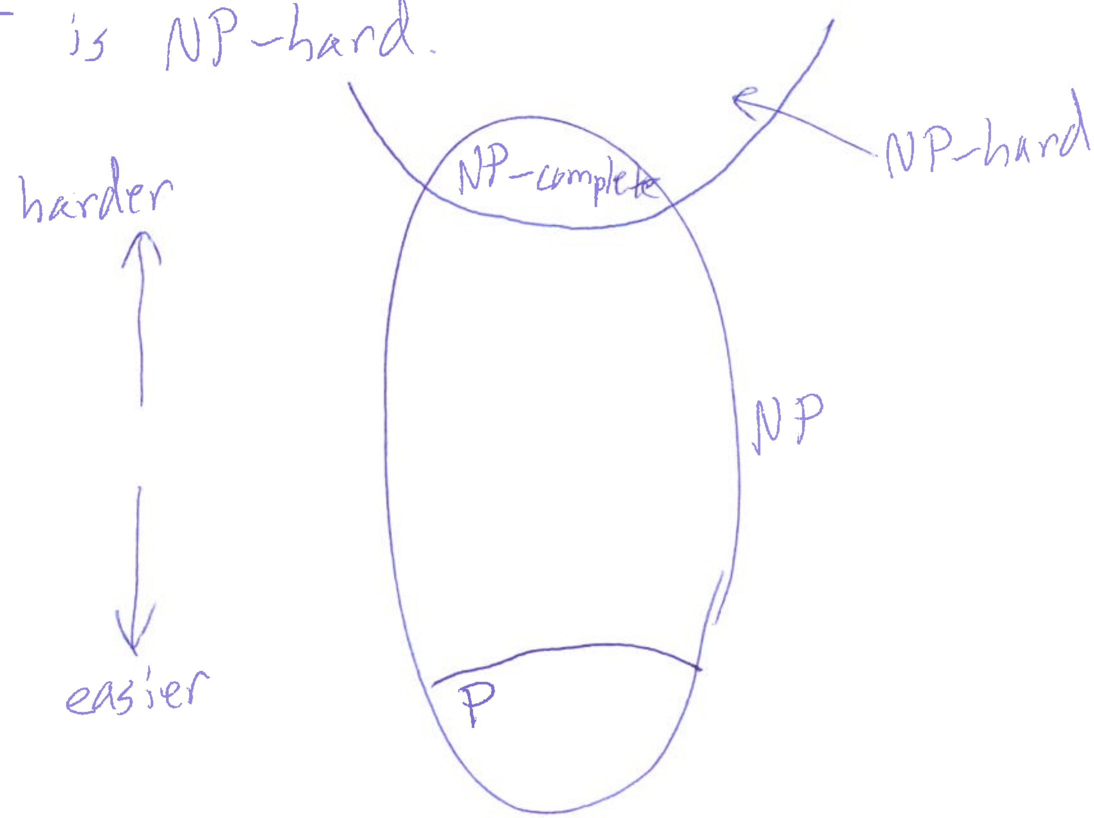
Prop: $\leq_p$ is reflexive: $(\Pi \underset{\text{identity}}{\leq_p} \Pi)$

and transitive: $(\Pi_1 \underset{f}{\leq_p} \Pi_2 \underset{g}{\leq_p} \Pi_3 \Rightarrow \Pi_1 \underset{g \circ f}{\leq_p} \Pi_3)$ $\blacksquare$

Def: (1) A decision problem Π is NP-hard if $\Pi_1 \leq_p \Pi$ for every $\Pi_1 \in NP$.

[Π is at least as hard as any NP problem.]

(2) Π is NP-complete if $\Pi \in NP$ and Π is NP-hard.



Def: $\Pi_1 \equiv_p \Pi_2$ (Π_1 & Π_2 are p-equivalent) if $\Pi_1 \leq_p \Pi_2$ and $\Pi_2 \leq_p \Pi_1$. Equivalence relation.

Prop: The NP-complete problems are an equivalence class under $\equiv_p$.

Proof: (1) Suppose Π_1 & Π_2 are NP-complete

Then $\Pi_1 \in NP$ and Π_2 is NP-hard

$\therefore \Pi_1 \leq_p \Pi_2$ by definition.

Similarly $\Pi_2 \leq_p \Pi_1$ $\quad \therefore \Pi_1 \equiv_p \Pi_2$

(2) WTS If Π_1 is NP-complete and $\Pi_1 \equiv_p \Pi_2$, then Π_2 is NP-complete:

Know $\Pi_1 \leq_p \Pi_2$ and Π_1 is NP-hard.

So, for any $\Pi \in NP$, $\Pi \leq_p \Pi_1$

So by transitivity, $\Pi \leq_p \Pi_2$. $\therefore \Pi_2$ is NP-hard.

Finally: $\Pi_2 \leq_p \Pi_1$ and $\Pi_1 \in NP$

$\therefore \Pi_2 \in NP$ by ~~the~~ a previous proposition

$\therefore \Pi_2$ is NP-complete.

So — NP-complete problems form an equiv. class under $\equiv_p$. $\square$