Clase 7

Santiago Cifuentes

May 29, 2025

1. Considerar el siguiente prolema:

   - LEX-SAT-BIT $= \{\langle\varphi, i\rangle : \varphi$ es una fórmula satisfacible y la menor asignación que la satisface (donde menor se define usando el orden lexicográfico) fija la variable i en $1\}$

   Probar LEX-SAT-BIT $\in \mathsf{P}^{\mathsf{NP}}$. Argumentar por qué el problema no debería estar en NP.

2. Probar que $\mathsf{P}^{\mathsf{NP}} = \mathsf{P}^{\mathsf{coNP}} \subseteq \Sigma_2^p \cap \Pi_2^p$.

3. Probar que $\mathsf{E}^{\mathsf{E}} \neq \mathsf{E}$.

4. Probar que $\mathsf{NP}^{\mathsf{NP}\cap\mathsf{coNP}} = \mathsf{NP}$.

5. Dada una clase $\mathcal{C}$, se define $\texttt{low}(\mathcal{C}) = \{\Pi \subseteq \Sigma^\star : \mathcal{C}^\Pi = \mathcal{C}\}$. Probar que $\texttt{low}(\mathsf{NP}) = \mathsf{NP} \cap \mathsf{coNP}$.

Resolución
2) $\mathsf{P}^{\mathsf{NP}} = \mathsf{P}^{\mathsf{coNP}}$ es inmediato. Para la segunda parte, probemos que $\mathsf{P}^{\mathsf{NP}} \subseteq \Sigma_2^p$, la otra inclusión sale usando que P^{NP} está cerrado por complemento.
Si $\Pi \in \mathsf{P}^{\mathsf{NP}}$ hay una máquina M^{SAT} que resuelve Π en tiempo polinomial usando llamados SAT. Notemos que si M^{SAT} corre en $p(|x|)$, entonces hace a lo sumo $p(|x|)$ llamados, cada uno de tamaño a lo sumo $p(|x|)$.
Podemos definir una máquina M' como $M'(x, a_1, \dots, a_{p(|x|)}) = M(x)^{a_1,\dots,a_{p(|x|)}}$ donde M' simula M pero usando de respuestas a $a_1, \dots, a_{p(|x|)}$. Está claro que si $M^{\text{SAT}}(x) = 1$ entonces existen algunos $a_1, \dots, a_{p(|x|)}$ tales que $M'(x, a_1, \dots, a_{p(|x|)}) = 1$. La vuelta también vale siempre y cuando los $a_1, \dots, a_{p(|x|)}$ se correspondan con las consultas que hace la simulación. Esto se puede escribir con la siguiente fórmula:

$x \in \Pi \iff M(x) = 1 \iff$
$\exists a_1, \ldots, a_{p(|x|)}, \varphi_1, \ldots \varphi_{p(|x|)}, c_1^+, \ldots, c_{p(|x|)}^+ \forall c_1^- \ldots c_{p(|x|)}^-$
$M'(x, a_1, \ldots, a_{p(|x|)}) = 1 \wedge$
$\bigwedge_{i=1}^{p(|x|)} \varphi_i$ es la query i que hace la simulación $M'(x, a_1, \ldots, a_{p(|x|)})$
$\bigwedge_{i=1}^{p(|x|)} \big((a_i = 1 \implies M_{\text{SAT}}(\varphi_i, c_i^+) = 1) \wedge (a_i = 0 \implies M_{\text{SAT}}(\varphi_i, c_i^-) = 0)\big)$

donde M_{SAT} es el verificador de SAT. Los cuantificadores adivinan una cantidad polinomial de bits, y el predicado se puede evaluar en tiempo polinomial.
3) Probemos que $2\mathsf{E} \subseteq \mathsf{E}^{\mathsf{E}}$, lo cual implica que E está incluida estricamente en E^{E}. Para eso, tomemos $\Pi \in 2\mathsf{E}$. Vale que $\Pi \in \mathrm{TIME}[2^{2^{n^k}}]$ para algún k usando algún algoritmo M. Luego,

$$\Pi_{pad} = \{x01^{2^{|x|^k}} : x \in \Pi\}$$

está en E, pues se resuelve parseando en lineal y llamando a M, lo cual toma tiempo $O(2^{2^{|x|^k}}) = O(2^{|y|})$, que es exponencial en el tamaño de entrada. Ergo Π se puede resolver en E^{E} escribiendo, dado x, $|x|01^{2^{n^k}}$ y luego llamando a Π_{pad} como oráculo.
También se puede probar que $\mathsf{E}^{\mathsf{E}} \subseteq 2\mathsf{E}$ simulando la máquina directamente: si la máquina oracular corre en 2^{n^k} y el oráculo es un problema que se resuelve en 2^{n^c}, entonces se hacen a lo sumo 2^{n^k} llamados donde cada uno toma a lo sumo $2^{2^{n^k c}} = 2^{2^{cn^k}}$.