Clase 2

Santiago Cifuentes

April 10, 2025

1. Probar que los siguientes lenguajes están en NP.

   - CLIQUE $= \{\langle G, k\rangle : G$ tiene una clique de tamaño $k\}$
   - COMPOSITE $= \{\langle c, l, r\rangle : c$ tiene un factor f que cumple $l \leq f \leq r\}$
   - SUBSET-SUM $= \{\langle X, T\rangle$: existe un subconjunto $S \subseteq X$ tal que $\sum_{x\in S} x = T\}$

2. Explicar por qué no es fácil probar que los siguientes problemas no están en NP.

   - DIOFANT $= \{\langle F(x_1, \ldots, x_k)\rangle : F$ es un multinomio con alguna raíz natural$\}$
   - UNSAT $= \{\langle \varphi(x_1, \ldots, x_n)\rangle : \varphi$ es una fórmula proposicional insatisfacible $\}$
   - SMALLEST $= \{\langle \varphi\rangle$: no existe ninguna fórmula $\psi < \varphi$ tal que $\varphi(\mathbf{x}) = \psi(\mathbf{x})$ para todo $x\}$

3. Probar que NP $\subseteq$ EXP.

4. Sea $\mathrm{NP}_{O(\log n)}$ la clase definida de forma idéntica a NP, pero pidiendo que el certificado tenga tamaño $O(\log n)$. Probar que $\mathrm{P} = \mathrm{NP}_{O(\log n)}$.

5. Probar que el modelo no determinístico es equivalente al modelo de certificados, incluso si permitimos que la bifurcación del modelo no determinístico sea polinomial.

6. Reducir INDSET a CLIQUE y a VERTEX-COVER, definido como

   - VERTEX-COVER $= \{\langle G, k\rangle : G$ tiene un vertex cover[1] de tamaño menor o igual a $k\}$

7. Probar que todo problem en NP se reduce a HALT. Generalizar a cualquier problema computable, y concluir que HALT es R-completo.

[1] Dado un grafo $G = (V, E)$ un vertex cover $C \subseteq V$ es un subconjunto de los nodos tal que para toda arista $vw \in E$ vale que $v \in C$ o $w \in C$.

Resolución

1. Vamos por problemas:

   - Para CLIQUE pedimos como certificado la clique en si. Está claro que tiene tamaño menor o igual a $|V| \log |V|$. El certificador debe revisar que tenga tamaño mayor o igual a k y que todos los nodos estén conectados.
   - Para COMPOSITE pedimos un número f entre l y r que sea factor de c. Vale que $|l| \leq |f| \leq |r| \leq |c|$, por lo que es polinomial. El certificador tiene que revisar que el número esté entre l y r y que divida a n. Para esto último puede usar el algoritmo de división de la primaria, que corre en tiempo polinomial en función de la cantidad de dígitos.
   - Para SUBSET-SUM alcanza con pedir que nos indiquen un subconjunto de X. Esto requiere $|X|$ bits. El certificador solo chequea que la suma de esos elementos sea k.

2. Vamos por problemas:

   - Para DIOFANT no podemos pedir la raíz porque no tenemos control de su tamaño. Es más, no hay ninguna función computable que nos diga qué tan grande es la raíz si existe, debido a que el problema es indecidible.
   - Para UNSAT es fácil verificar la opuesta, pero no la afirmativa. Si se pudiera, NP = coNP, lo cual se cree que es falso.
   - Para SMALLEST podemos pedir de certificado la fórmula más chica si la hay, pero no podemos chequear esto último. Más adelante vamos a ver que el problema está en Π_2^p, aunque no se sabe si es hard.

3. Visto en clase, con fuerza bruta probamos todos los certificados. Si tienen tamaño $p(n)$ y el verifador corre en $q(n)$, entonces el algoritmo tiene complejidad $O(2^{p(n)} q(p(n)))$.

4. El algoritmo anterior es polinomial si reemplazamos $p(n) \leq c \log(n)$.

5. Demostrado en clase, solo repasar la equivalencia.

6. Son inmediatas observando que si I es un conjunto independiente de G entonces $V \setminus I$ es un vertex cover de G y a la vez una clique de G^c.

   Para probar que CLIQUE es NP-completo solo falta ver que es hard (que está en NP ya lo sabemos por el primer ejercicio). La reducción que proponemos es $f(\langle G, k \rangle) = \langle \overline{G}, k \rangle$. Es polinomial porque se puede complementar un grafo en $O(n^2)$. Para la correctitud, notemos que $\langle G, k \rangle$ es una instancia positiva de INDSET si y solamente si existe un subconjunto I de los nodos de G tales que $|I| \geq k$ y I es un conjunto independiente. Por las observaciones anteriores esto vale si y solamente si I es una

clique de tamaño mayor o igual a k en $\overline{G}$, y esto ocurre por definición si $\langle \overline{G}, k \rangle \in$ CLIQUE.

Para vertex cover se puede tomar $f(\langle G, k \rangle) = \langle G, n - k \rangle$.

7. Si $\Pi \in$ NP existe una máquina M tal que $x \in \Pi \iff \exists c \in \Sigma^{p(n)} M(x, c)$. Dado el código de M puedo escribir el código de la máquina del ejercicio 3 (no ejecutarlo, solo escribirlo), y agregar que si no encuentra certificado no pare. Este programa M' tiene un número asociado z, y cumple que $x \in \Pi \iff M_z(z)$ para (el programa ignora la entrada). Luego, tengo una reducción a HALT. En general, dado un problema Π computable hay una máquina que lo resuelve M, y podemos seguir la misma estrategia. Con recursivamente enumerable es análogo.