CURSO DE COMPLEJIDAD COMPUTACIONAL 2021

Práctico 4

Temas: La jerarquía polinomial (PH). Ejemplos de clases de PH. Colapso de la jerarquía polinomial. Completitud en las clases de PH. Máquinas con oráculo.

4.1. Ejercicio 5.11 de Arora-Barak.

SUCCINT SET-COVER =

$\{\langle S=\{\varphi_1,\varphi_2...\varphi_m\},k\rangle : \exists S'\subseteq\{1..m\}, |S'|\le k, \phi=\vee_{i\in S}\varphi_i$ es una tautología$\}$.

Cada φ_i es una fórmula booleana 3-DNF (disyunctive normal form, es un AND de cláusulas con 3 OR's). k es un entero positivo.

Probar que SUCCINT SET-COVER $\in \Sigma_2^p$. Especificar en que consisten los certificados asociados a los cuantificadores existenciales y universales de la clase.

4.2. Sea el lenguaje

$L=\{\langle \phi,1^B\rangle : \phi \in$ DNF y $\exists\ \varphi\in$ CNF equivalente a ϕ con a lo sumo B cláusulas$\}$

DNF: Conjunto de fórmulas booleanas en Disjunctive Normal Form, en donde cada fórmula es una disjunción de cláusulas, cada una de las cuales es una conjunción de literales booleanos.

CNF: Conjunto de fórmulas booleanas en Conjunctive Normal Form, en donde cada fórmula es una conjunción de cláusulas, cada una de las cuales es una disjunción de literales booleanos.

Probar que $L\in\Sigma_2^p$.

4.3. Ejercicio 5.12 de Arora-Barak.

Probar que si $\Sigma_i^p=\Pi_i^p$ entonces $\mathrm{PH}=\Sigma_i^p$ (la jerarquía colapsa al i-ésimo nivel)

4.4. Ejercicio 5.3 de Arora-Barak.

Mostrar que si $3SAT$ tiene una reducción polinomial de Karp a $\overline{3SAT}$, entonces se verifica PH=NP.

4.5. Ejercicio 5.1 de Arora-Barak.

Probar que si $\Sigma_i SAT$ es completo para Σ_i^p bajo reducciones de tiempo polinomial.

4.6. Consideramos el lenguaje

$\exists USAT=\left\{\varphi : \exists\, x\in\{0,1\}^n\ \exists!\, y\in\{0,1\}^m\ \varphi(x_1,...,x_n,y_1,...,y_m)=1\right\}$

donde φ es una fórmula booleana y la notación "$\exists!$" significa "existe exactamente uno".

Por ejemplo $\varphi(x,y_1,y_2)=\left(x\wedge y_1\wedge\overline{y_2}\right)\vee\left(\overline{x}\wedge y_1\right)\in\exists USAT$ porque la asignación $x=1$ hace que la única asignación para y que hace verdadera la fórmula es $y_1=1$ y $y_2=0$.

Probar que $\exists USAT$ es Σ_2^p - completo.

4.7. Mostrar que si $NP = NP^{NP}$ entonces se cumple $coNP \subseteq NP$

4.8. (Reducción de un problema de búsqueda a un problema de decisión para isomorfismo de grafos). Sea A un algoritmo que resuelve el problema de isomorfismo de grafos, es decir dados dos grafos G_1 y G_2 devuelve "si"si los grafos son isomorfos y "no"en caso contrario. Mostrar que existe una máquina con oráculo polinomial B con entrada $< G_1 , G_2 >$ tal que B^A devuelve un isomorfismo entre G_1 y G_2 si existe alguno y "no son isomorfos"en caso contrario.