MA50B Complejidad Computacional **18 de Mayo, 2002**

Pauta Control 2

Profesor: M. Kiwi **Auxiliar: E. Moreno**

Problema 1:

(i).- Para $\omega \in \{0,1\}^*$ denotaremos por $N(\omega)$ al entero no–negativo cuya representación binaria es ω. Consideremos el lenguaje $D \subseteq \{0,1\}^*$ tal que $\omega \in D$ si y sólo si $M_{N(\omega)}$ rechaza ω.

Veamos que D es decidible. Para ello, consideremos un enumerador E de L (su existencia esta garantizada por el hecho que L es reconocible). Sin pérdida de generalidad podemos asumir que la i–ésima salida producida por E es $\langle M_i \rangle$, pues de lo contrario basta reindexar los elementos de L para que así sea. Definimos ahora la mT N tal que en la entrada ω simula E hasta producir la $N(\omega)$-ésima salida $\langle M \rangle$ y luego simula M en la entrada ω. La máquina N acepta ω si y sólo si M rechaza ω.

Veamos que N decide D. Como E es enumerador, eventualmente produce la $N(\omega)$-ésima codificación $\langle M \rangle$ de una mT que se detiene en todas sus entradas. En particular M se detiene en ω. Por lo tanto, N se detiene cualquiera sea su entrada. Además, la $N(\omega)$-ésima salida $\langle M \rangle$ de E corresponde a $M_{N(\omega)}$ que al simularse en la entrada ω acepta si y sólo si $\omega \notin D$. Es decir, N decide D.

Finalmente, probemos que $D \neq L_i$ cualquiera sea $i \in \mathbb{N}$. En efecto, si $D = L_i$ y ω fuese la representación en binario de i, tendríamos que N acepta ω si y sólo si M_i rechaza ω. Es decir, $\omega \in D$ si y sólo si $\omega \notin L_i$, lo que constituye una contradicción.

(ii).- Dado G grafo no–dirigido, basta ciclar sobre los $\binom{|V(G)|}{k}$ posibles conjuntos de k nodos de G y verificar si alguno de ellos es un clique en G. El ciclo es de largo a lo más $|V(G)|^k$ y el proceso de verificación toma tiempo $O(k^2)$ en una RAM, suponiendo que la entrada G está codificada vía una matriz de adyacencia. Luego, en una RAM, el mencionado procedimiento se puede implementar en tiempo $O(|V(G)|^k)$, lo que si bien es ineficiente para cualquier k que no sea muy pequeño, es un polinomio en el tamaño de la entrada. Como lo que se puede hacer en una RAM en tiempo polinomial, también se puede hacer en una mT en tiempo polinomial, concluimos que $CLIQUE_k \in P$.

(iii).- Sea ϕ una fórmula Booleana en forma 2 conjuntiva normal sobre las variables $x_1, \ldots, x_n$. Sea G_ϕ el grafo dirigido con $2n$ nodos que llamaremos $x_1, \ldots, x_n, \bar{x}_1, \ldots, \bar{x}_n$ y tal que si $u \vee v$ es una clausula de ϕ, entonces $(\bar{u}, v)$ y $(\bar{v}, u)$ son arcos de $E(G_\phi)$.

Afirmamos que $\langle \phi \rangle \notin 2SAT$ si y sólo si existe i tal que en G_ϕ hay un ciclo que contiene a x_i y a $\bar{x}_i$. En efecto, supongamos que hay un ciclo en G_ϕ que contiene a x_i y a $\bar{x}_i$. Luego, existen nodos $v_1, \ldots, v_t, v'_1, \ldots, v'_s$ tales que $x_i, v_1, \ldots, v_t, \bar{x}_i, v'_1, \ldots, v'_s$ es un ciclo en G_ϕ. Por definición de G_ϕ, sigue que $x_i \Rightarrow v_1 \Rightarrow \ldots \Rightarrow v_t \Rightarrow \bar{x}_i \Rightarrow v'_1 \Rightarrow \ldots \Rightarrow v'_s \Rightarrow x_i$. En particular $x_i \Leftrightarrow \bar{x}_i$. No existe ninguna asignación de valores de verdad para la cual esta equivalencia pueda cumplirse si y sólo si $\langle \phi \rangle \notin 2SAT$. Luego, para decidir si $\langle \phi \rangle \in 2SAT$ hay que determinar si existe algún i para el cual G_ϕ tiene un ciclo que contiene a x_i y $\bar{x}_i$ — para ello basta ciclar sobre los $i \in \{1, \ldots, n\}$ y ver si hay caminos en G_ϕ de x_i a $\bar{x}_i$ y de $\bar{x}_i$ a x_i. Como determinar si un nodo t de un grafo G es alcanzable desde otro nodo s se puede decidir en tiempo polinomial, sigue que $2SAT \in$ P.

(iv).- Un certificado de pertenencia de $\langle a_1, \ldots, a_n, B, k \rangle$ en $BIN{-}PACK$ es una colección $S_1, \ldots, S_l$, $l \leq k$, partición de $\{1, \ldots, n\}$. El proceso de verificación consiste en comprobar que para todo $1 \leq i \leq l$, se tiene que $\sum_{j \in S_i} a_j \leq B$. Claramente, una mTN puede generar el mencionado certificado y realizar el proceso de verificación en tiempo polinomial. Sigue que $BIN{-}PACK \in$ NP.

La completitud la estableceremos vía la restricción de $BIN{-}PACK$ a $PARTICION$. Para ello, restringimos las instancias $\langle a_1, \ldots, a_n, B, k \rangle$ a aquellas en que $\sum_{i=1}^{n} a_i$ es par, $B = \sum_{i=1}^{n} a_i/2$ y $k = 2$. Dichas instancias están en $BIN{-}PACK$ si y sólo si $\langle a_1, \ldots, a_n \rangle$ está en $PARTICION$. Sigue que $PARTICION \leq_P BIN{-}PACK$.

(v).- Un certificado de pertenencia de $\langle G \rangle$ en $MEDIO{-}CLIQUE$ es una lista de nodos $u_1, \ldots, u_m \in V(G)$ donde $m = |V(G)|/2$. Para verificar que dicho conjunto de nodos induce un clique en G basta comprobar que $(u_i, u_j) \in E(G)$ para todo $i \neq j$. Claramente, una mTN puede generar el mencionado certificado y realizar el proceso de verificación en tiempo polinomial. Sigue que $MEDIO{-}CLIQUE \in$ NP

La completitud la estableceremos mostrando que $CLIQUE \leq_P MEDIO{-}CLIQUE$. En efecto dado $G = (V, E)$ grafo y $0 \leq k \leq |V(G)|$, construimos $G' = (V', E')$ tal que $V \subseteq V'$. Entre todo par de nodos de $V' \setminus V$ agregamos un arco y también agregamos arcos entre todos los nodos de $V' \setminus V$ y V. Es fácil ver que G tiene un clique de tamaño al menos k si y sólo si G' tiene un clique de tamaño $k + |V' \setminus V| = k + |V'| - |V|$. Luego, eligiendo V' de forma que $k + |V'| - |V| \geq |V'|/2$, por ejemplo tomando $|V'| = 2(|V| - k)$, sigue que

$$\langle G, k \rangle \in CLIQUE \iff \langle G' \rangle \in MEDIO{-}CLIQUE\,. \qquad (1)$$

La construcción es factible si $|V'| = 2(|V| - k) \geq |V|$, i.e., si $|V| \geq 2k$. Si este no es el caso, se obtiene G' agregando $2k - |V|$ nodos y ningún arco al grafo G. Es fácil comprobar que también en este caso se cumplirá (**??**).

Es fácil ver que la construcción que a G y k le asocia G' es a tiempo polinomial.

Problema 2:

(i).- Observar que $\langle\phi(x_1,\ldots,x_n)\rangle \in MIN{-}FORM$ si y sólo si para todo $\psi(x_1,\ldots,x_n)$ fórmula Booleana tal que $|\langle\psi\rangle| < |\langle\phi\rangle|$ existen $a_1,\ldots,a_n$ tales que $\phi(a_1,\ldots,a_n) \neq \psi(a_1,\ldots,a_n)$.

Sea entonces A la mTA tal que en la entrada $\omega = \langle\phi(x_1,\ldots,x_n)\rangle$ procede como se describe a continuación:

1. En sus primeros $|\omega|-1$ pasos ocupa estados universales para adivinar (y escribir en una de sus cintas) una secuencia de símbolos (del alfabeto en que se representan las codificaciones $\langle\cdot\rangle$).
2. Verifica que la adivinanza realizada en el paso anterior representa una codificación de una fórmula Booleana ψ en las variables $x_1,\ldots,x_n$.
3. En los siguientes n pasos ocupa estados existenciales para adivinar (y nuevamente guardar en una de sus cintas) valores $a_1,\ldots,a_n \in \{0,1\}$.
4. Acepta si $\phi(a_1,\ldots,a_n) \neq \psi(a_1,\ldots,a_n)$.

Claramente A tiene dos niveles de alternancia y como los dos procesos de verificación recién mencionados son fácilmente realizables en tiempo polinomial, obtenemos la conclusión deseada.

(ii).- La demostración está basada en la típica simulación de espacio por tiempo. Consiste en hacer una búsqueda en profundidad en el árbol de cómputo de una mTA A a espacio $S(n)$.

Para determinar si el árbol de cómputos asociados a A es tal que A acepta, se determina el valor de la raíz de dicho árbol vía el siguiente procedimiento recursivo:

- Una hoja del árbol es de aceptación si esta asociada a una configuración cuyo estado es de aceptación. De lo contrario, la hoja es de rechazo.
- Un nodo interno universal es de aceptación si y sólo si todos sus hijos son de aceptación.
- Un nodo interno existencial es de aceptación si y sólo si alguno de sus hijos es de aceptación.

Claramente, A acepta si y sólo si la raíz de su árbol asociado es de aceptación.

Observar ahora que al igual que una mTN clásica con similares restricciones de espacio, la máquina A en la entrada ω tiene a lo más

$$(|\omega| + O(1)) \left(S(|\omega|)|\Sigma_A|^{S(|\omega|)}\right)^{O(1)} ,$$

configuraciones posibles. Luego, recurriendo a la típica técnica de aumentar A con un contador, sin pérdida de generalidad, podemos asumir que la profundidad de todas las ramas de cálculo de A es a lo más $S(|\omega|)$. Sigue que la búsqueda en profundidad del árbol de cómputo de A toma tiempo $2^{O(S(n))}$. Por lo tanto todo el proceso de evaluación del valor de sus nodos toma también $2^{O(S(n))}$.

(iii).- La primera inclusión está basada en la típica simulación de tiempo por espacio. Consiste en hacer una búsqueda en profundidad en el árbol de cómputo de una mTA A a tiempo $T(n)$ y observar que ello requiere espacio a lo más $T(n)$ para recordar cuál rama del árbol se está explorando y espacio $T(n)$ para simular A en dicha rama.

La segunda inclusión es consecuencia directa de la demostración del Teorema de Savitch. En efecto, dicha demostración garantiza que conocida una mT M a espacio $T(n)$ que decide un lenguaje L, se puede construir dado ω, en tiempo polinomial una fórmula Booleana totalmente cuantificada ψ_ω tal que $|\langle\psi_\omega\rangle| = O(T^2(|\omega|))$ y ψ_ω es verdadera si y sólo si M acepta ω (i.e, $\omega \in L$). Como una fórmula Booleana totalmente cuantificada es fácilmente evaluable por una mTA en tiempo lineal, se obtiene la conclusión deseada.

(iv).- Por inducción en k. Si $k = 1$, entonces $\Sigma_k^P = \text{NP}$ y $\Pi_k^P = \text{coNP}$. Una mTA para la cual todos sus estados no–deterministas son del tipo existencial es simplemente una mTN. Por lo tanto, $L \in \text{NP}$ si y sólo si L puede ser decidido por una mTA a tiempo polinomial cuyos estados no–deterministas son todos del tipo existencial. Si ahora consideramos $L \in \text{coNP}$, tenemos que $\overline{L}$ puede ser decidido por una mTA A a tiempo polinomial cuyos estados no–deterministas son todos del tipo existencial. Sea A' la mTA que se obtiene al hacer los estados de aceptación (respectivamente rechazo) de A, estados de rechazo (respectivamente aceptación) y todos sus estados no–deterministas, estados del tipo universal. Es fácil ver que A' es una mTA a tiempo polinomial sin estados existenciales y que decide L. Concluimos la base de la inducción observando que una mTA que tiene un sólo tipo de estados no–deterministas, existenciales o universales, es una mTA con 1 nivel de alternancia.

Supongamos entonces que la afirmación es cierta para k. Sea $C \in \Sigma_{k+1}^P$, sea M la mTN con oráculo $L \in \Pi_k^P$ y a tiempo polinomial que decide C. Por inducción, existe una mTA A que decide L en tiempo polinomial con a lo más k niveles de alternancia. Sea entonces A' la mT que realiza los siguientes pasos en la entrada ω:

1. Adivina, usando estados existenciales, las preguntas $\omega_1, \ldots, \omega_m$ y respuestas $r_1, \ldots, r_m$ que M le hace a su oráculo en ω.

2. Simula M en ω. Cuando M haga la i–ésima consulta a su oráculo, A verifica que esta pregunta es ω_i. Si no es así, rechaza. De lo contrario sigue simulando M como si la respuesta hubiese sido r_i.

3. Si al simular M esta no realiza exactamente m consultas a su oráculo, A rechaza.

4. La máquina A acepta si M acepta dadas las respuestas $r_1, \ldots, r_m$ a sus consultas y

si se cumple que

$$(\forall i)(\omega_i \in L \text{ si y sólo si } r_i = q_s)\,. \tag{2}$$

Concluimos, por hipótesis inductiva, que $\omega_i \in L$ puede ser decidido por una mTA A' con k niveles de alternancia cuyas primeras transiciones no–deterministas son del tipo universal. Sigue que para que A pueda decidir si se cumple (**??**) basta con que en forma no–determinista y usando estados universales simule A' en la entrada $\omega_1, \ldots, \omega_m$ (cada ω_i en una rama distinta de su árbol de cómputo) y verifique que la respuesta es $r_1, \ldots, r_m$ respectivamente. Es fácil ver que A acepta si y sólo su entrada está en C. Claramente, A tiene $k+1$ niveles de alternancia, sus primeras transiciones no–deterministas son del tipo existencial y es a tiempo polinomial.