**Ma50b Complejidad Computacional** **20 de Abril de 2013**

# Pauta Control 1

*Prof. Cátedra: M. Kiwi* *Prof. Auxiliar: P. Muñoz, D. Salas*

PROBLEMA 1:

(i).- Sea $A = (\Sigma, Q, s, \delta, F)$ autómata finito determinista que reconoce $L$. Construimos un nuevo autómata $A'$ cuyo conjunto de estados es $Q \cup (Q \times \Sigma)$, su estado inicial es $s$, su conjunto de estados de aceptación es $F$, y su función de transición $\delta'$ esta dada por:

- $\delta'(q,a) = (q,a)$ para todo $q \in Q$ y $a \in \Sigma$.
- $\delta'((q,a),b) = \delta(\delta(q,b),a)$ para todo $q \in Q$ y $a, b \in \Sigma$.

Afirmamos que $A'$ reconoce $\mathrm{Swap}(L)$. En efecto, basta establecer, por inducción, que para todo $n$, el autómata $A'$ en $\omega_2\omega_1 \ldots \omega_{2n}\omega_{2n-1}$ alcanza el estado $q$ si y solo si $A$ en $\omega_1 \ldots \omega_{2n}$ alcanza el estado $q$.

(ii).- Un autómata finito no-determinista que reconoce $L$ solo tiene que "adivinar" el $i$ adecuado, y luego, utilizando 4 estados, contar si a partir del $i$-ésimo caracter de la entrada aparecen cuatro 1's más que 0's — por ejemplo, como en el autómata finito no-determinista $A$ de la Figura 1.

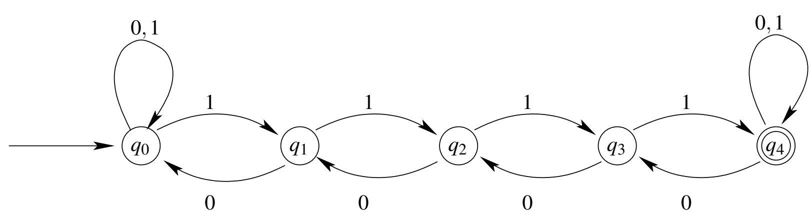


Figura 1: Diagrama de transiciones del autómata finito $A$.

Para probar que $A$ reconoce $L$ basta mostrar, por inducción, que $A$ en $\omega_1 \ldots \omega_n$ alcanza el estado $q_r$ si existen $1 \leq i \leq j \leq n$ tales que

$$\sum_{s=i+1}^{j} \omega_s = r + \sum_{s=i+1}^{j} (1 - \omega_s).$$

(iii).- Sea $A$ un autómata finito determinista que decide $L$. Sea $A'$ con los mismos estados y transiciones de $A$ excepto porque se han removido de su diagrama de transición las transiciones que "salen" de los estados en $F_A$. Afirmamos que $L_{A'} = \mathrm{NPR}(L)$. Para ello, basta probar por inducción que $A'$ en $\omega_1 \ldots \omega_n$ alcanza el estado $q$ si y solo si $A$ en $\omega_1 \ldots \omega_n$ alcanza el estado $q$ y que para todo $1 \leq i < n$ en $\omega_1 \ldots \omega_i$ no alcanza ningún estado en $F_A$.

PROBLEMA 2:

(i).- Las propiedades de reflexividad y simetría se tienen por definición de $\equiv_L$. Falta establecer la transitividad. Sean entonces $x \equiv_L y$ e $y \equiv_L w$. Por definición, para todo $z \in \Sigma^*$ se tiene que $(xz, yz \in L \vee xz, yz \notin L)$ y $(yz, wz \in L \vee yz, wz \notin L)$. Consideremos dos casos. Si $yz \in L$, entonces se debe tener que $xz, wz \in L$. Si $yz \notin L$, entonces $xz, wz \notin L$. En cualquier caso, tenemos que $xz, wz \in L$ o $xz, wz \notin L$, es decir $x \equiv_L z$.

(ii).- Supongamos que $L$ es regular y que $A$ es un autómata finito determinista que lo reconoce. Afirmamos que el índice de $L$ es a lo más el número de estados de $A$. En efecto, sean $x, x' \in \Sigma_L^*$ pertenecientes a clases de equivalencia distintas, i.e. $x \not\equiv_L x'$. Sean $q$ y $q'$ los estados alcanzados por $A$ al procesar $x$ y $x'$ respectivamente. Si $q = q'$, entonces para todo $z \in \Sigma^*$ se tendría que $A$, al procesar $xz$ y $x'z$, alcanzaría el mismo estado. En particular, $x \equiv_L x'$, contradicción. Se tiene entonces que $q \neq q'$ y que el número de clases de equivalencia distintos es a lo más $|Q_A|$.

Supongamos ahora que el índice de $A$ es finito. Sigue que $\equiv_L$ particiona $\Sigma^*$ en una cantidad finita de clases de equivalencia, digamos $[x_1], \ldots, [x_s]$. Construiremos un autómata finito determinista $A$ con $s$ estados, uno por cada clase de equivalencia. El estado de partida de $A$ será la clase de equivalencia a la que pertenece la palabra vacía $\varepsilon$. Los estados finales de aceptación serán todas aquellas clases de equivalencia que contengan elementos de $L$. Finalmente, la función de transición $\delta$ de $A$ será tal que $\delta([x], a) = [xa]$. Veamos que la función de transición queda bien definida. En efecto, si $x \equiv_L x'$, entonces para todo $z \in \Sigma^*$, en particular para $z = az'$ con $a \in \Sigma$, sabemos que $xaz', x'az' \in L$ o $xaz', x'az' \notin L$. En particular, $xa \equiv_L x'a$ cualquiera sea $a \in \Sigma$.

Afirmamos que $A$ reconoce $L$. Por construcción, si $\omega \in L$, entonces $A$ termina en el estado $[\omega]$, que es de aceptación dado que en particular contiene en particular a $\omega$. Sigue que $\omega$ es aceptado por $A$. Supongamos ahora que $A$ acepta $\omega$. Por definición de estado de aceptación, se tendrá que existe algún $\omega' \in L$ tal que $\omega \equiv_L \omega'$. Luego, $\omega z, \omega' z \in L$ o $\omega z, \omega' z \notin L$, cualquiera sea $z$, en particular se tiene para $z = \varepsilon$. Sigue que $\omega, \omega' \in L$ o $\omega, \omega' \notin L$. Como $\omega' \in L$, debe tenerse el primer caso, es decir $\omega \in L$.

(iii).- Afirmamos que si $n, m \in \mathbb{N}$, $n \neq m$, entonces $0^n \not\equiv_L 0^m$, de donde sigue que el índice de $L$ es no acotado. Para probar la afirmación, procedamos por contradicción. Supongamos que $0^n \equiv 0^m$. Entonces, para todo $z \in \{0,1\}^*$ se tiene que $xz, yz \in L$ o $xz, yz \notin L$. En particular, lo anterior es válido para $z = 1^n$, por lo que $0^n 1^n, 0^m 1^n \in L$ o $0^n 1^n, 0^m 1^n \notin L$, lo cual no es cierto, obteniéndose así la contradicción buscada.

PROBLEMA 3:

(i).- Dado $S$ lenguaje, definimos el lenguaje $S^{\mathcal{R}}$ como aquel cuyos elementos son los $\omega^{\mathcal{R}}$ tales que $\omega \in S$. Notar que por definición de $L$, tenemos que

$$L = \left\{ \langle A \rangle : A \text{ es autómata finito determinista y } L_A \cap L_A^{\mathcal{R}} \neq \emptyset \right\}.$$

Dado $A = (\Sigma, Q, s, \delta, F)$ autómata finito no-determinista, sea $A^{\mathcal{R}}$ el autómata finito con $\varepsilon$-transiciones con alfabeto $\Sigma$, conjunto de estados $Q \cup \{q_{ac}\}$, conjunto de estados finales $\{s\}$, estado de partida $\{q_{ac}\}$, y función de transición $\delta^{\mathcal{R}}$ dada por

- $\delta^{\mathcal{R}}(q_{ac}, \varepsilon) = F$.
- $q' \in \delta^{\mathcal{R}}(q, a)$ si y solo si $\delta(q', a) = q$.

Es fácil ver que $A$ acepta $\omega$ si y solo si $A^{\mathcal{R}}$ acepta $\omega^{\mathcal{R}}$, por lo que $A^{\mathcal{R}}$ reconoce $L_A^{\mathcal{R}}$. Por construcción vista en clases, sigue que existe un autómata finito $A'$ tal que $L_{A'} = L_A \cap L_A^{\mathcal{R}}$. Implícitamente, hemos establecido que la reducción $f(\cdot)$ que a $\langle A\rangle$, con $A$ autómata finito determinista, le asocia $f(\langle A\rangle) = \langle A'\rangle$ es una función totalmente recursiva tal que

$$\langle A\rangle \in L \Longleftrightarrow f(\langle A\rangle) \in \overline{V}_{AFD},$$

i.e. $L \leq_m \overline{V}_{AFD}$ donde $V_{AFD} = \{\langle A'\rangle : A'$ autómata finito determinista tal que $L_{A'} = \emptyset\}$.

En clase auxiliar se probó que $V_{AFD}$ es decidible. Como los lenguajes decidibles son cerrados bajo completación, $\overline{V}_{AFD}$ también es decidible. Como $L$ mucho-a-uno reduce a un lenguaje decidible, por resultado conocido, se tiene que $L$ es decidible.

(ii).- Bastará probar que $A_{mT}$ mucho-a-uno reduce a $L$. Consideremos una instancia $\langle M, \omega\rangle$ de $A_{mT}$. Sea $M'$ la máquina de Turing con alfabeto de entrada $\Sigma_M \cup \{\#\}$, $\# \notin \Sigma_M$, tal que

- En la entrada $\omega\#$, $\omega \in \Sigma^*$, simula $M$ en $\omega$ y acepta/rechaza de acuerdo a si $M$ acepta/rechaza.
- En la entrada $\#\omega$, $\omega \in \Sigma^*$, acepta.
- En cualquier otra entrada no cubierta por alguno de los casos anteriores, rechaza.

Observemos que si $M$ acepta $\omega$, entonces $M'$ acepta $\omega' = \omega\#$ y también acepta $\omega'^{\mathcal{R}} = \#\omega^{\mathcal{R}}$. En otras palabras, $\langle M', \omega'\rangle \in L$. Si $M$ no acepta $\omega$, entonces $M'$ no acepta $\omega' = \omega\#$, pero sí acepta $\omega'^{\mathcal{R}} = \#\omega^{\mathcal{R}}$. Equivalentemente, $\langle M', \omega'\rangle \notin L$.

Es fácil ver que la transformación que a $\langle M, \omega\rangle$ le asocia $\langle M', \omega\#\rangle$ es totalmente recursiva. En resumen, hemos probado que $A_{mT} \leq_m L$.

(ii.1).- Dado $M_i$ máquina de Turing tal que $\langle M_i\rangle \in A$, sea $M'_{i,j}$ una máquina de Turing cuyo diagrama de transiciones es idéntico al de $M_i$ pero con $j$ estados adicionales, ninguno de ellos accesibles desde alguno de los estados en $Q_{M_i}$ y todos tales que en dicho estado, leyendo cualquier símbolo, la máquina escribe el mismo símbolo leído y mueve su cabeza lectora a la derecha. Claramente, $L_{M'_{i,j}} = L_{M_i}$ cualquiera sea $j$. Usando la convención estándar de codificación de máquinas de Turing, tenemos que $|\langle M'_{i,j}\rangle| \geq |\langle M_i\rangle| + j$.

Dado que $A$ es reconocible, existe un enumerador $E_A$ para $A$. Consideremos la siguiente máquina de Turing:

$T$ = En $\langle M\rangle$ donde $M$ es máquina de Turing,
(1) Simular $E_A$. Por cada salida $\langle M_i\rangle$ generada por $E_A$:
(2) Obtener el diagrama de estados de $M$ alcanzables a partir de $s_M$.
(3) Obtener el diagrama de estados de $M_i$ alcanzables a partir de $s_{M_i}$.
(4) Si los diagramas obtenidos son iguales, aceptar.

Sea $B$ el lenguaje reconocido por la máquina de Turing $T$ (en particular, $B$ es reconocible). Notar que $\langle M'_{i,j}\rangle \in B$ para todo $i, j \in \mathbb{N}$. Veamos que $B$ cumple con las propiedades deseadas. En efecto, si $\langle M_i\rangle \in A$ y $C > 0$, entonces $L_{M'_{i,C}} = L_{M_i}$, $|\langle M'_{i,C}\rangle| \geq |\langle M_i\rangle| + C$, y $\langle M'_{i,C}\rangle \in B$. Si $\langle M'\rangle \in B$, entonces existe un $i \in \mathbb{N}$ tal que $E_A$ genera $\langle M_i\rangle$ (i.e. $\langle M_i\rangle \in A$) y $L_{M'} = L_{M_i}$.

(ii.2).- Sea $T'$ una máquina de Turing que en la entrada $\langle M\rangle$, $M$ máquina de Turing, actúa como la máquina $T$ de la parte (ii.1) precedente, excepto porque en el paso (1) se simula un enumerador $E_B$ del lenguaje $B$ cuya existencia esta garantizada por la parte (ii.1) anterior, y a lo más hasta que el enumerador $E_B$ genere $m = |\langle M\rangle|$ salidas. Si $T'$ no acepta para ninguna de las $m$ salidas, entonces $T'$ rechaza.

Sea $D$ el lenguaje reconocido por $T'$. Como $T'$ se detiene cualquiera sea su entrada, se tiene que $D$ es decidible. Veamos que $D$ tiene la propiedad deseada. En efecto, si $\langle M \rangle \in D$, entonces existe un $\langle M' \rangle \in B$ tal que $M'$ y $M$ tienen el mismo diagrama de estados no aislados, en particular $L_{M'} = L_M$. Pero si $\langle M' \rangle \in B$, sabemos que existe $\langle M'' \rangle \in A$ tal que $L_{M''} = L_{M'} = L_M$. Supongamos ahora que $\langle M_i \rangle \in A$. Sigue que $\langle M_{i,j} \rangle \in B$ cualquiera sea $j \in \mathbb{N}$. Sea $t \in \mathbb{N}$ la primera vez que el enumerador $E_B$ genera una salida en $\{\langle M_{i,j} \rangle : j \in \mathbb{N}\}$, digamos $\langle M_i' \rangle$. Como hay una infinidad de $M_{i,j}$'s con $j$ tomando valores en $\mathbb{N}$, necesariamente debe existir un $j_0$ tal que $|\langle M_{i,j_0} \rangle| \geq t$. Se tiene que $T'$ en la entrada $\langle M_{i,j_0} \rangle$ necesariamente acepta, porque eventualmente $E_B$ generará la salida $\langle M_i' \rangle$ con el mismo diagrama de estados no aislados que $M_{i,j_0}$. Luego, $L_{M_i} = L_{M_i'} = L_{M_{i,j_0}}$ y $\langle M_{i,j_0} \rangle \in D$.