Automi e Linguaggi **(M. Cesati)**

Facoltà di Ingegneria, Università degli Studi di Roma Tor Vergata

Compito scritto del 21 giugno 2019

Esercizio 1 [6] Siano $L_1, L_2 \subseteq \Sigma^*$ linguaggi regolari; dimostrare che

$$L_1 \setminus L_2 = \{w \in L_1 \,|\, w \notin L_2\}$$

è un linguaggio regolare.

Soluzione: La dimostrazione è immediata considerando che $L_1 \setminus L_2 = L_1 \cap \overline{L_2}$ e che la classe dei linguaggi regolari è chiusa rispetto alle operazioni di complemento ed intersezione.

Un'altra dimostrazione consiste nel derivare un DFA N che riconosce il linguaggio differenza a partire dai DFA $M = (Q, \Sigma, \delta, q_0, F)$ e $M' = (Q', \Sigma, \delta', q_0', F')$ che riconoscono rispettivamente L_1 e L_2. In particolare, $N = (Q \times Q', \Sigma, \delta'', (q_0, q_0'), F'')$, ove la funzione di transizione $\delta'' : Q \times Q' \times \Sigma \to Q \times Q'$ è definita come $\delta''(q, q', x) = (\delta(q, x), \delta'(q', x))$, e l'insieme di stati di accettazione è $F'' = \{(q, q') \,|\, q \in F, q' \notin F'\}$.

Esercizio 2 [6] Sia $A = \left\{ww' \,|\, w, w' \in \{0,1\}^*, w' = \overline{w}^{\mathcal{R}}\right\}$, ($w'$ è la stringa ottenuta da w rovesciando l'ordine dei bit e complementandone il valore). Dimostrare che A è un CFL esibendo una grammatica opportuna.

Soluzione: La più semplice CFG G che genera le stringhe di A è:

$$S \to 0S1 \,|\, 1S0 \,|\, \epsilon$$

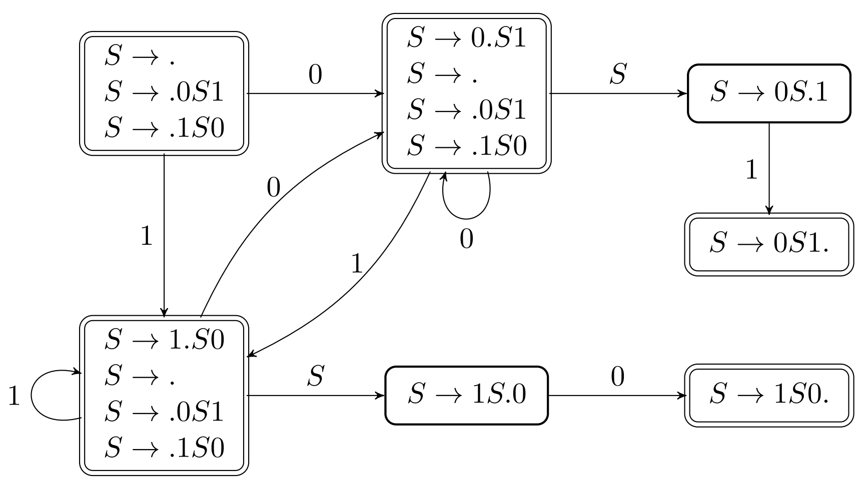
Infatti è immediato verificare che ogni stringa terminale generata da G è costituita da un numero pari di bit, quindi può essere decomposta in due sottostringhe w e w' di pari lunghezza. Il primo bit di w e l'ultimo bit di w' devono essere diversi perché generati nell'applicazione di una singola regola $0S1$ o $1S0$. Allo stesso modo, il secondo bit di w ed il penultimo bit di w' devono essere differenti perché generati nell'applicazione di un'altra regola, e così via. Formalmente, ogni derivazione da S lunga un passo che genera una stringa terminale deve produrre $\epsilon \in A$. Sia dunque assunto per vero che ogni derivazione da S lunga n passi che genera una stringa terminale produca un elemento di A. In una derivazione da S lunga $n+1$ passi che genera una stringa terminale z deve iniziare applicando una regola $S \to xS\overline{x}$, e continuare derivando da S una stringa terminale in n passi. Per ipotesi induttiva dunque $z = xv\overline{v}^{\mathcal{R}}\overline{x} = w\overline{w}^{\mathcal{R}} \in A$.

Viceversa, consideriamo una qualunque stringa binaria $w\overline{w}^{\mathcal{R}}$, e dimostriamo per induzione sulla lunghezza della stringa w che può essere generata da G. Se $|w| = 0$, l'asserto segue dalla regola $S \to \epsilon$. Supponiamo che l'asserto sia vero per $|w| \le i$, e consideriamo la stringa $w\overline{w}^{\mathcal{R}}$, ove $|w| = i+1$. Si ha dunque che $w\overline{w}^{\mathcal{R}} = xv\overline{v}^{\mathcal{R}}\overline{x}$, con $x \in \{0,1\}$. Per costruzione, è possibile derivare da S la stringa non terminale $xS\overline{x}$; per ipotesi induttiva, è possibile derivare da S la stringa $v\overline{v}^{\mathcal{R}}$, poiché $|v| = i$; pertanto, la grammatica può generare tutte le stringhe $w\overline{w}^{\mathcal{R}}$ arbitrariamente lunghe.

Esercizio 3 [7] Dimostrare che la grammatica ottenuta nel precedente esercizio è non deterministica.

Soluzione: Intuitivamente, ogni grammatica che genera le stringhe del linguaggio $A = \left\{ww' \mid w, w' \in \{0,1\}^*, w' = \overline{w}^{\mathcal{R}}\right\}$ deve essere nondeterministica perché un PDA che riconosce le stringhe del linguaggio deve necessariamente utilizzare il nondeterminismo per indovinare l'occorrenza nella sequenza di input del primo simbolo di $w' = \overline{w}^{\mathcal{R}}$.

Formalmente, eseguiamo il DK-test sulla grammatica G, ottenendo:



Poiché almeno uno degli stati terminali contiene un "dot" seguito da un simbolo terminale, il test è fallito e quindi la grammatica è non deterministica.

Esercizio 4 [10] Sia $B = \{ww' \mid w, w' \in \{0,1\}^*, w' = \overline{w}\}$, ($w'$ è la stringa ottenuta da w complementando ogni bit). Dimostrare che B non è un CFL.

Soluzione: Per assurdo, supponiamo che B sia CFL e che quindi valga per esso il "Pumping lemma". Sia dunque p la "pumping length" per B. In questo esercizio la scelta della stringa che contraddice le conclusioni del lemma deve essere fatta con attenzione. Ad esempio, la stringa $0^p1^p1^p0^p$ non andrebbe bene, perché può essere pompata ponendo $u = 0^{p-1}$, $v = 0$, $x = \epsilon$, $y = 1$, $z = 1^{2p-1}0^p$: infatti, $uv^ixy^iz = 0^{p-1}0^i1^i1^{2p-1}0^p = 0^{p+i-1}1^p1^{p+i-1}0^p$.

Consideriamo invece la stringa $s = 0^p1^p01^p0^p1$: evidentemente $s \in B$ (in quanto $\overline{0^p1^p0} = 1^p0^p1$), quindi deve essere possibile determinare una suddivisione $s = uvxyz$ tale che per ogni $i \geq 0$, $uv^ixy^iz \in B$, $|vy| > 0$ e $|vxy| \leq p$. Distinguiamo i seguenti casi:

1. Se vy non contiene esattamente lo stesso numero di zero ed uno, pompando verso il basso o verso l'alto si otterrebbe una stringa con una eccedenza di zero od uno, che dunque non potrebbe far parte di B. Ciò implica anche che $|vy| > 1$.

2. Se la sottostringa vxy fosse interamente nella prima metà di s, allora pompando verso il basso si otterrebbe una stringa uxz in cui uno dei bit della seconda occorrenza di 1^p in s finirebbe nell'ultima posizione della prima metà di uxz (in quanto $|vxy| \leq p$ e $|vy| > 1$). Pertanto uxz non può essere in B perché il bit in ultima posizione della prima metà deve essere 0.

3. Analogamente, la sottostringa vxy non può essere interamente nella seconda metà di s, altrimenti pompando verso il basso uno dei bit della prima occorrenza di 1^p in s finirebbe nell'ultima posizione della prima metà di uxz, e ciò significa che tale stringa non potrebbe far parte di B perché essa termina con 1.

4. Pertanto, vxy deve contenere sia un bit della prima metà di s che un bit della seconda metà. Poichè però vy deve contenere un ugual numero di zero ed uno, l'unica possibilità è che vy sia o la stringa 01 oppure la stringa 10, ove lo 0 è quello in ultima posizione della prima metà di s. Pompando verso il basso si ottiene una stringa in cui nell'ultima posizione della prima metà vi è un 1, dunque tale stringa non può far parte di B.

Poiché non è possibile suddividere la stringa s in accordo al pumping lemma, concludiamo che il pumping lemma non può essere applicato, e pertanto che l'ipotesi che B fosse un CFL è falsa.

Esercizio 5 [11] Il problema Dominating Set è il seguente: dato un grafo non diretto $G = (V, E)$, ed un numero intero k, determinare se esiste un sottoinsieme di nodi $V' \subseteq V$ di cardinalità k tale che ogni nodo del grafo o appartiene a V' oppure è adiacente ad un nodo in V' (o entrambe le cose). Dimostrare che Dominating Set è NP-completo descrivendo una riduzione polinomiale da Vertex Cover.

Soluzione: Per prima cosa dimostriamo che Dominating Set (DS) appartiene a NP. Il problema è polinomialmente verificabile perché ogni istanza che fa parte del linguaggio ha come certificato il sottoinsieme di nodi del grafo che costituisce il dominating set: è certamente di dimensione non superiore al numero di nodi del grafo, e verificare che ogni nodo del grafo fa parte od è adiacente al sottoinsieme può essere facilmente realizzato da un algoritmo che esegue in tempo polinomiale nella dimensione dell'istanza del problema:

M= "On input $\langle G=(V,E),k,V'\rangle$, where G is a graph, $V' \subseteq V$:

1. if $|V'| > k$: reject
2. for each node v in V:
    3. if $v \in V'$ then continue with next node in step 1
    4. for each edge $e \in E$:
        5. if e links v to a node in V', continue with next node in step 1
    6. Reject, because node v is not dominated by V'
7. Accept, because all nodes in V are dominated by V'"

Il numero totale di passi eseguiti dall'algoritmo è $O(n\,m\,n) = O(n^4)$, ove $n = |V(G)|$ e $m = |E(G)| = O(n^2)$.

Consideriamo ora una riduzione polinomiale da Vertex Cover (VC) a DS. Sia $(G = (V,E),k)$ una istanza di VC. Costruiamo un nuovo grafo $G' = (V', E')$ in questo modo: $V' = V \cup V_E$ è costituito dai nodi di G e da un nodo v_e per ciascun arco $e \in E$; in totale quindi $|V'| = n + m = |V| + |E|$. $E' = E \cup E_V$ è costituito dagli archi di G e, per ciascun nodo $v_e \in V_E$, da due archi (v_e, v) e (v_e, w) ove $e = (v,w)$; in totale quindi $|E'| = 3m = 3\,|E|$. Possiamo dimostrare che $(G,k) \in$ VC se e solo se $(G', k+s) \in$ DS, ove s è il numero di nodi $S \subseteq V$ "isolati" (senza archi incidenti).

Supponiamo che $(G,k) \in$ VC; dunque esiste $U \subseteq V$ tale che $|U| = k$ ed ogni arco di G è incidente ad almeno un nodo di U. Consideriamo il sottoinsieme di nodi $U' = U \cup S$ in G' (esiste perché tutti i nodi di G sono anche nodi di G'), e dimostriamo che è un dominating set di dimensione $k+s$. Infatti, sia $x \in V(G') = V \cup V_E$: se $x \in V$, allora o $x \in S$, e dunque $x \in U'$, oppure esiste un arco $e \in E$ incidente su x. Poiché U è un ricoprimento degli archi in G, esiste un nodo $y \in U$ tale che $e = (x,y)$; pertanto, il nodo x è dominato dal nodo $y \in U'$.

Se invece $x \in V_E$, allora $x = v_e$ per un certo arco $e \in E$: dunque, U deve contenere un nodo y che ricopre e; allora, per costruzione di E_V, $(y, v_e) \in E_V$, e quindi x è dominato da $y \in U'$.

Per la direzione opposta, supponiamo che $(G', k + s) \in$ DS, e quindi esiste un sottoinsieme U', con $|U'| = k + s$, che domina ogni nodo di G'. Risulta evidente che $S \subseteq U'$, perché l'unico modo per dominare nodi isolati è inserirli nel sottoinsieme dominante. Un'altra osservazione è che se U' contiene un qualunque nodo $v_e \in V_E$, è possibile sostituire in U' il nodo v_e con uno qualunque dei due nodi v, w tali che $e = (v, w)$. Infatti per costruzione di G' il nodo v_e può dominare solo se stesso, v e w; ma v (o equivalentemente w) domina almeno se stesso, w e v_e, quindi sostituendo v_e con v si ottiene un dominating set di dimensione pari od inferiore a quella di U' che domina almeno lo stesso insieme di nodi di G'. Consideriamo dunque il dominating set U'' in cui ogni nodo v_e è stato sostituito da un nodo in V come appena descritto, e sia $U = U'' \setminus S$. Il sottoinsieme U ha cardinalità $\leq k$, è un sottoinsieme di nodi di G, ed è un ricoprimento degli archi in G. Infatti, sia $e \in E$, con $e = (v, w)$. Poiché il nodo v_e deve essere dominato da qualche nodo in U'', per costruzione v, w, od entrambi appartengono al sottoinsieme dominante U''. Naturalmente $v, w \notin S$, quindi v, w oppure entrambi, appartengono ad U. Pertanto, l'arco e risulta ricoperto da un elemento di U.

Abbiamo dunque dimostrato che la trasformazione da (G, k) a $(G', k + s)$ è una riduzione tra problemi. È inoltre evidente che tale trasformazione può essere costruita in tempo polinomiale. Pertanto, VC $\leq_m$ DS, e quindi DS è NP-hard in quanto VC è NP-completo. Ciò conclude la dimostrazione di NP-completezza di DS.