STRUMENTI FORMALI PER LA BIOINFORMATICA

Automi finiti - Parte 4, Algoritmo di Brzozowski

Questi trasparenti sono basati sulla traduzione in italiano di alcuni trasparenti in inglese gentilmente resi disponibili dal Prof. Hendrik Jan Hoogeboom, LIACS, Università di Leiden (Olanda), su argomenti tratti dal libro

Jeffrey Shallit:
Second Course in Formal Languages and Automata Theory,
Cambridge University Press, 2009.

Dato un DFA $M = (Q, \Sigma, \delta, q_0, F)$, sia M^R la macchina ottenuta invertendo gli archi nel diagramma di stato di M.

Dato un DFA $M = (Q, \Sigma, \delta, q_0, F)$, sia M^R la macchina ottenuta invertendo gli archi nel diagramma di stato di M.

Formalmente, definiamo

$$M^R = (Q, \Sigma, \delta^R, F, \{q_0\})$$

dove

$$\delta^R(q', a) = \{q \in Q \mid \delta(q, a) = q'\}$$

Dato un DFA $M = (Q, \Sigma, \delta, q_0, F)$, sia M^R la macchina ottenuta invertendo gli archi nel diagramma di stato di M.

Formalmente, definiamo

$$M^R = (Q, \Sigma, \delta^R, F, \{q_0\})$$

dove

$$\delta^R(q', a) = \{q \in Q \mid \delta(q, a) = q'\}$$

Nota che M^R non è un NFA poiché può avere più di uno stato iniziale. Ma è chiaro come estendere la definizione di accettazione in questo NFA generalizzato.

È possibile definire una trasformazione da M^R a un DFA equivalente mediante la stessa costruzione vista per trasformare un NFA in un DFA **(subset construction)**.

È possibile definire una trasformazione da M^R a un DFA equivalente mediante la stessa costruzione vista per trasformare un NFA in un DFA **(subset construction)**.

L'unica differenza significativa è che lo stato iniziale del DFA corrispondente è l'insieme degli stati iniziali dell'NFA generalizzato.

La macchina M^R

È possibile definire una trasformazione da M^R a un DFA equivalente mediante la stessa costruzione vista per trasformare un NFA in un DFA **(subset construction)**.

L'unica differenza significativa è che lo stato iniziale del DFA corrispondente è l'insieme degli stati iniziali dell'NFA generalizzato.

Mancano inoltre le transizioni con etichetta ϵ.

Per un NFA generalizzato $\mathcal{A}$ sia $S(\mathcal{A})$ il DFA ottenuto attraverso la costruzione del DFA equivalente (subset construction), **usando solo gli stati raggiungibili dallo stato iniziale/dagli stati iniziali**.

Per un NFA generalizzato $\mathcal{A}$ sia $S(\mathcal{A})$ il DFA ottenuto attraverso la costruzione del DFA equivalente (subset construction), **usando solo gli stati raggiungibili dallo stato iniziale/dagli stati iniziali**.

Teorema (Brzozowski)

Dato un DFA M, la macchina

$$S((S(M^R))^R)$$

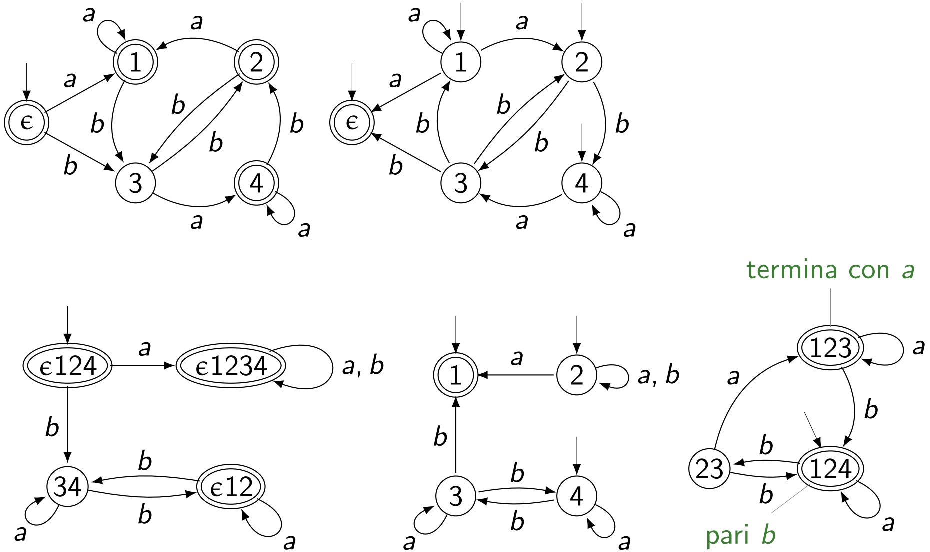
è un DFA minimale equivalente ad M.

Brzozowski provò che è possibile ottenere l'automa minimale eseguendo le operazioni seguenti due volte: inversione (reverse) e poi determinizzazione.

Brzozowski provò che è possibile ottenere l’automa minimale eseguendo le operazioni seguenti due volte: inversione (reverse) e poi determinizzazione.

È quasi magico che funzioni.

Esempio: minimizzazione di Brzozowski



- La complessità degli algoritmi sugli automi è in genere valutata in funzione del numero n degli stati dell'automa e della cardinalità k dell'alfabeto degli input.

- La complessità degli algoritmi sugli automi è in genere valutata in funzione del numero n degli stati dell'automa e della cardinalità k dell'alfabeto degli input.
- Sono stati ottenuti raffinamenti dell'algoritmo $\mathcal{A}lg$ per individuare gli stati distinguibili:
  J. Hopcroft, 1971, complessità $\mathcal{O}(kn \lg n)$;
  A. Valmari, 2012, complessità $\mathcal{O}(n + m \lg m)$, con m numero delle transizioni. Se l'automa verifica alcune ipotesi aggiuntive, l'algoritmo di Valmari ha complessità $\mathcal{O}(n + m \lg n)$.

- La complessità degli algoritmi sugli automi è in genere valutata in funzione del numero n degli stati dell'automa e della cardinalità k dell'alfabeto degli input.
- Sono stati ottenuti raffinamenti dell'algoritmo $\mathcal{A}lg$ per individuare gli stati distinguibili:
  J. Hopcroft, 1971, complessità $\mathcal{O}(kn \lg n)$;
  A. Valmari, 2012, complessità $\mathcal{O}(n + m \lg m)$, con m numero delle transizioni. Se l'automa verifica alcune ipotesi aggiuntive, l'algoritmo di Valmari ha complessità $\mathcal{O}(n + m \lg n)$.
- L'algoritmo di Brzozowski ha una complessità $\mathcal{O}(kn\, 2^{2n})$, quindi esponenziale. Ma è eccellente per le computazioni "a mano" su automi di piccole dimensioni e in pratica sembra essere non troppo lento.