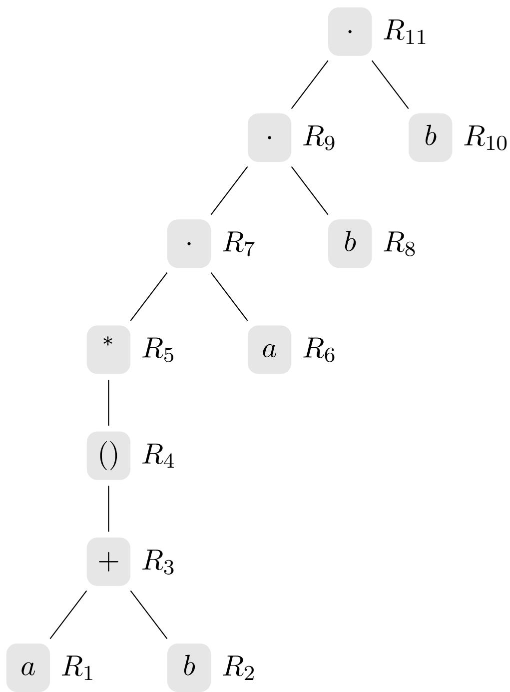
Azzolini Riccardo 2020-11-12

Dalle espressioni regolari agli ϵ-NFA — Esempio e complessità

1 Esempio di costruzione

Si vuole costruire l'ϵ-NFA corrispondente all'espressione regolare $R = (a + b)^*abb$. La struttura di quest'espressione può essere rappresentata come un albero, in cui le foglie sono le espressioni elementari (i casi basi della definizione di espressione regolare) e i nodi interni sono gli operatori (i casi induttivi della definizione):



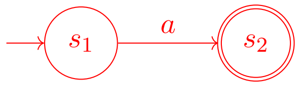
Siccome l'algoritmo di costruzione dell'ϵ-NFA opera in modo induttivo (costruendo prima gli automi per le sottoespressioni, e poi "combinandoli" nell'automa per l'intera espressione), esso corrisponde a una visita in *postordine* di quest'albero:

- prima si visita ricorsivamente il sottoalbero sinistro;
- poi si visita ricorsivamente il sottoalbero destro;

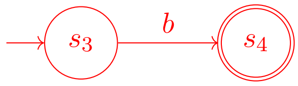
- infine si visita la radice.

Nella raffigurazione dell'albero riportata sopra, R_i indica l'i-esima sottoespressione che si incontra in questo ordine di visita. Dunque, bisogna costruire prima l'automa per R_1, poi quello per R_2, e così via, fino ad arrivare a $R_{11} = R$. In seguito, verranno illustrati tutti i passi di costruzione, evidenziando ogni volta in rosso i nuovi stati e le nuove transizioni.

1. Per l'espressione $R_1 = a$ si definisce il seguente automa A_{R_1}, secondo uno dei casi base dell'algoritmo di costruzione:

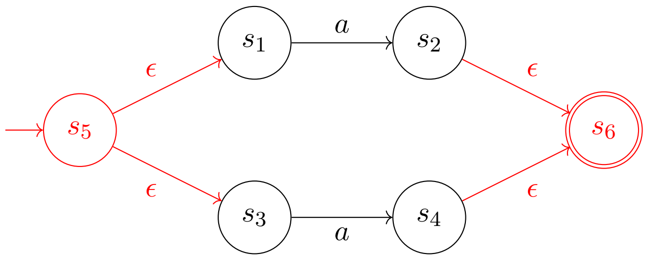


2. Analogamente, per $R_2 = b$ si costruisce il seguente A_{R_2}:

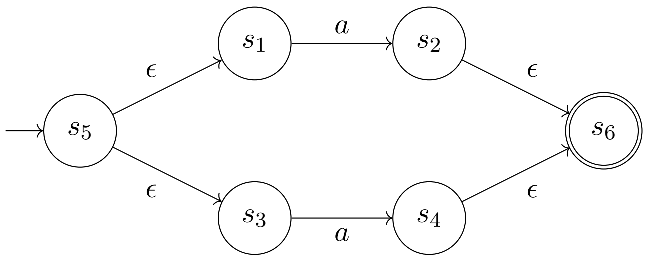


   Agli stati aggiunti in questo passo sono stati dati dei nomi *nuovi*, diversi da quelli usati in A_{R_1}, per evitare "conflitti" di nomi quando A_{R_1} e A_{R_2} automi verranno poi messi insieme. Lo stesso criterio verrà usato per scegliere i nomi degli stati aggiunti anche in tutti i passi successivi.

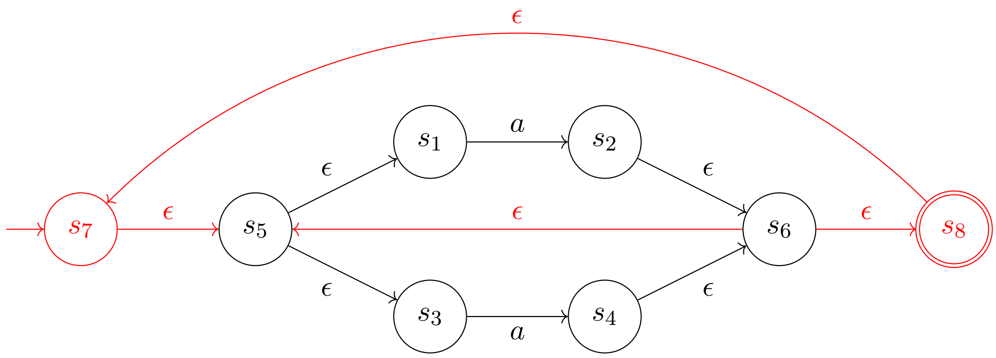
3. Per $R_3 = a + b$, si costruisce A_{R_3} in base a uno dei casi induttivi dell'algoritmo:



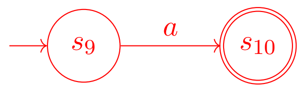
4. L'espressione $R_4 = (a + b)$ è semplicemente R_3 racchiusa tra parentesi, cioè genera lo stesso linguaggio di R_3, quindi anche l'automa rimane uguale, $A_{R_4} = A_{R_3}$:



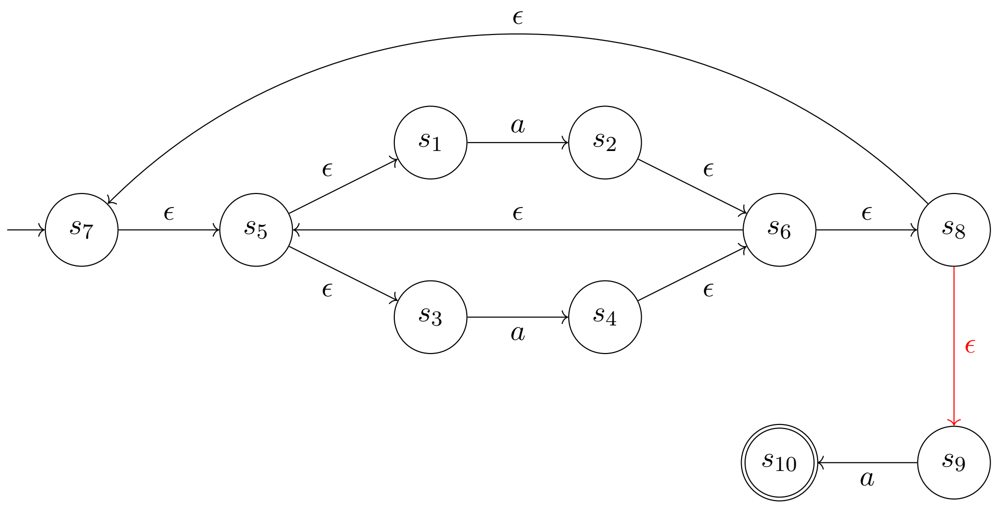
5. Per $R_5 = (a + b)^*$, A_{R_5} è:



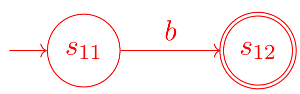
6. $R_6 = a$ è uguale a $R_1 = a$, ma all'interno dell'espressione R queste due a sono istanze diverse, che hanno ruoli diversi. Allora, non si può riutilizzare A_{R_1}, perché così gli stati avrebbero gli stessi nomi: nel proseguimento della costruzione, gli automi corrispondenti alle due istanze di a finirebbero per "sovrapporsi", "collassare" in un unico automa, mentre si vuole che rimangano separati. Bisogna invece costruire un automa A_{R_6}, i cui stati abbiano nomi nuovi:



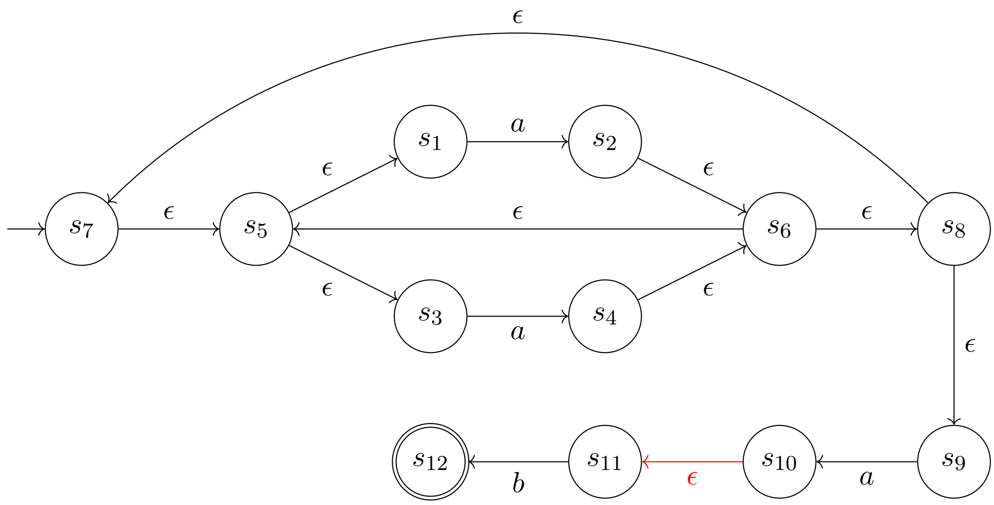
7. Per $R_7 = (a + b)^*a$ si definisce il seguente A_{R_7}:



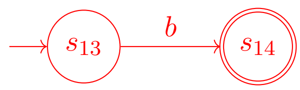
8. Anche per $R_8 = b$ bisogna costruire un nuovo automa A_{R_8}, invece di riutilizzare A_{R_2}:



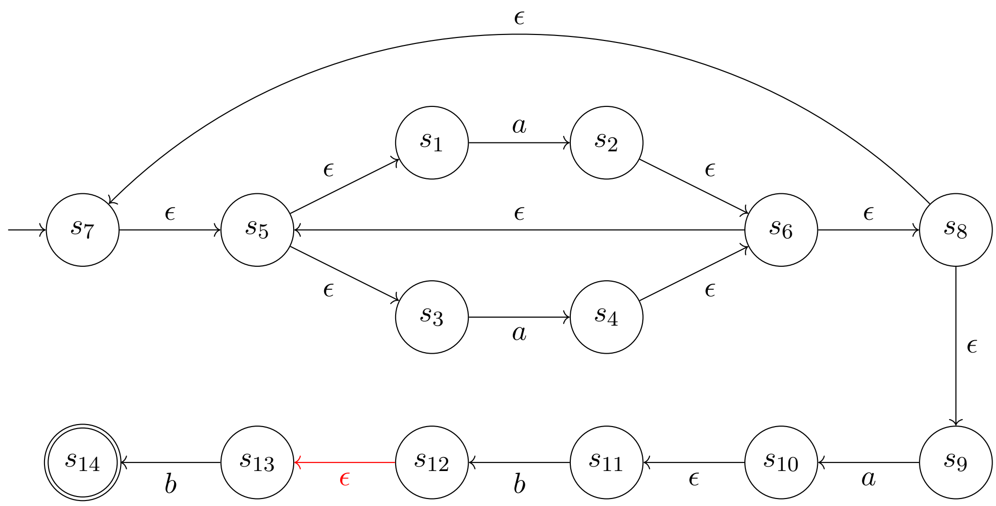
9. Per $R_9 = (a + b)^*ab$, A_{R_9} è:



10. Per $R_{10} = b$, si deve costruire di nuovo un automa $A_{R_{10}}$ che sia indipendente da A_{R_2} e A_{R_8}:



11. Infine, l'automa $A_{R_{11}}$ per l'intera espressione regolare $R_{11} = R = (a + b)^+ abb$ è:



2 Complessità dell'algoritmo

Data un'espressione regolare R, si definisce la sua *lunghezza* $|R|$ come il numero di simboli ($\varnothing$, ϵ, $a \in \Sigma$ e operatori, comprese le concatenazioni implicite) da cui essa è formata.

La costruzione dell'automa A_R corrispondente all'espressione R avviene in $|R|$ passi. A ogni passo vengono aggiunti al più due nuovi stati, quindi il numero complessivo di stati di A_R è $n \leq 2 \cdot |R|$. Analogamente, le transizioni aggiunte a ogni passo sono al massimo 4 (nei casi $R_1 + R_2$ e R^*), quindi il numero finale di transizioni presenti in A_R è $m \leq 4 \cdot |R|$. Segue che la dimensione dell'automa risultante è $O(|R|)$.

Usando le opportune strutture dati per la rappresentazione dell'automa, si ha poi che ogni passo di costruzione richiede un tempo proporzionale al numero di stati e transizioni nuovi aggiunti, quindi anche il tempo di costruzione di A_R è $O(|R|)$.