ELEMENTI DI INFORMATICA TEORICA

Parte 2: Linguaggi Formali e Automi

2.2 Automi a Stati Finiti

Giovanni Amendola

Corso di laurea triennale in Informatica
Università della Calabria

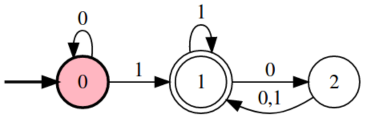
9 aprile 2022

Anno Accademico 2021/2022

Da ASF a grammatica

Possiamo associare ad un ASF una grammatica (regolare)

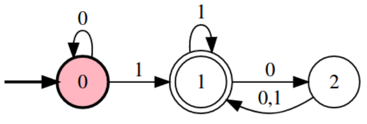
Esempio (Consideriamo l'ASF $\mathcal{A}$ del primo esempio)



Da ASF a grammatica

Possiamo associare ad un ASF una grammatica (regolare)

Esempio (Consideriamo l'ASF $\mathcal{A}$ del primo esempio)



La grammatica associata è

$$\mathcal{G}_{\mathcal{A}} = \langle \Sigma, Q, q_0, P \rangle$$

dove Σ è l'alfabeto, Q è il vocabolario dei simboli non terminali, q_0 è lo stato iniziale (corrispondente all'assioma S), e P le regole di produzione:

1. $q_0 \rightarrow 0q_0 \mid 1q_1 \mid 1$
2. $q_1 \rightarrow 1q_1 \mid 0q_2 \mid 1$
3. $q_2 \rightarrow 0q_1 \mid 1q_1 \mid 1 \mid 0$

corrispondenti agli stati di accettazione e alla matrice delle transizioni.

Da ASF a grammatica

Esempio (Analisi sintattica - *parsing*)

Consideriamo l'ASF $\mathcal{A}$ con $\mathcal{G}_{\mathcal{A}} = \langle \Sigma, Q, q_0, P \rangle$, dove P è dato da

1. $q_0 \to 0q_0 \mid 1q_1 \mid 1$
2. $q_1 \to 1q_1 \mid 0q_2 \mid 1$
3. $q_2 \to 0q_1 \mid 1q_1 \mid 1 \mid 0$

Verifichiamo che la stringa 1101 appartiene al linguaggio $\mathcal{L}(\mathcal{G}_{\mathcal{A}}) = \mathcal{L}(\mathcal{A})$.

Da ASF a grammatica

Esempio (Analisi sintattica - *parsing*)

Consideriamo l'ASF $\mathcal{A}$ con $\mathcal{G}_{\mathcal{A}} = \langle \Sigma, Q, q_0, P \rangle$, dove P è dato da

1. $q_0 \to 0q_0 \mid 1q_1 \mid 1$
2. $q_1 \to 1q_1 \mid 0q_2 \mid 1$
3. $q_2 \to 0q_1 \mid 1q_1 \mid 1 \mid 0$

Verifichiamo che la stringa 1101 appartiene al linguaggio $\mathcal{L}(\mathcal{G}_{\mathcal{A}}) = \mathcal{L}(\mathcal{A})$.

a. $1101 \leftarrow 110q_2$ (da 3)
b. $110q_2 \leftarrow 11q_1$ (da 2)
c. $11q_1 \leftarrow 1q_1$ (da 2)
d. $1q_1 \leftarrow q_0$ (da 1)

Automi non deterministici

- **Determinismo**: se un automa si trova in un certo stato e legge un certo simbolo, allora passa univocamente in un preciso stato.
- Gli ASF che abbiamo visto in precedenza sono deterministici.
- Più precisamente possiamo chiamarli **Automi a stati Finiti Deterministici** (AFD) o in inglese **Deterministic Finite state Automata** (DFA).

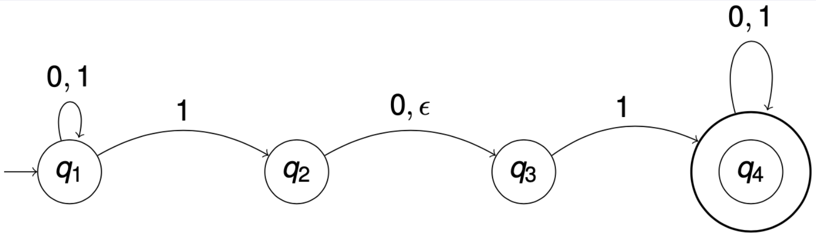
Automi non deterministici

- **Determinismo**: se un automa si trova in un certo stato e legge un certo simbolo, allora passa univocamente in un preciso stato.
- Gli ASF che abbiamo visto in precedenza sono deterministici.
- Più precisamente possiamo chiamarli **Automi a stati Finiti Deterministici** (AFD) o in inglese **Deterministic Finite state Automata** (DFA).
- **Non Determinismo**: data una configurazione possono esserci differenti possibili transizioni.
- Parleremo di **Automi a stati Finiti Non deterministici** (AFN) o in inglese **Non deterministic Finite state Automata** (NFA).
- Intuitivamente: avremo uno stato con due archi etichettati con lo stesso simbolo oppure uno stato con un arco etichettato con ϵ.

AFN (Automi a stati Finiti Non deterministici)

Esempio

La seguente rappresentazione è un AFN:



AFN (Automi a stati Finiti Non deterministici)

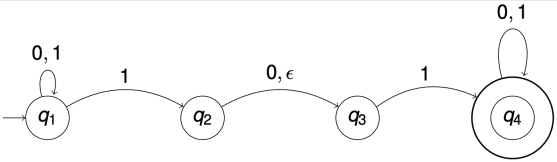
Definizione: AFN

Un *automa a stati finiti non deterministico* è una quintupla $\langle Q, \Sigma, \delta, q_0, F \rangle$ dove

1. Q è un insieme finito non vuoto di stati.
2. Σ è un alfabeto finito non vuoto.
3. $\delta : Q \times (\Sigma \cup \{\epsilon\}) \to \mathcal{P}(Q)$ è la funzione di transizione.
4. q_0 è lo stato iniziale.
5. $F \subseteq Q$ è l'insieme degli stati di accettazione.

AFN (Automi a stati Finiti Non deterministici)

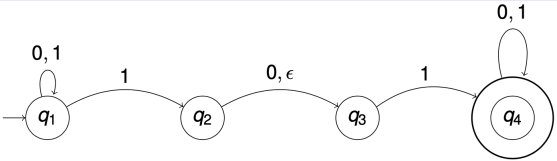
Esempio



Sia $\mathcal{N} = \langle Q, \Sigma, \delta, q_0, F \rangle$ dove

AFN (Automi a stati Finiti Non deterministici)

Esempio



Sia $\mathcal{N} = \langle Q, \Sigma, \delta, q_0, F \rangle$ dove

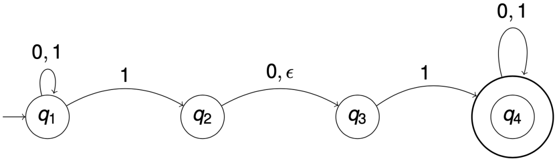
1. $Q = \{q_1, q_2, q_3, q_4\}$ è un insieme finito di stati
2. $\Sigma = \{0, 1\}$
3. $\delta : Q \times (\Sigma \cup \{\epsilon\}) \to \mathcal{P}(Q)$

	0	1	ϵ
q_1	$\{q_1\}$	$\{q_1, q_2\}$	$\emptyset$
q_2	$\{q_3\}$	$\emptyset$	$\{q_3\}$
q_3	$\emptyset$	$\{q_4\}$	$\emptyset$
q_4	$\{q_4\}$	$\{q_4\}$	$\emptyset$

4. q_1 è lo stato iniziale
5. $F = \{q_4\}$ è l'insieme degli stati di accettazione

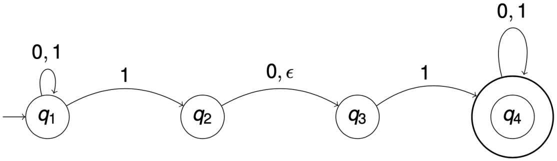
AFN (Automi a stati Finiti Non deterministici)

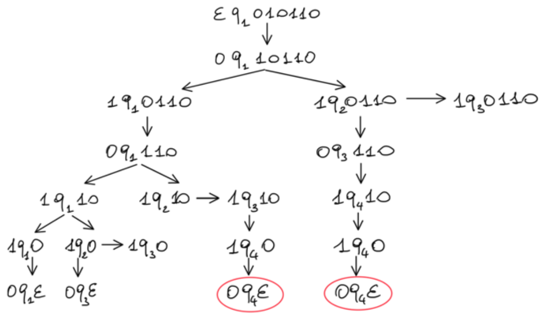
Esempio: Computazione della stringa 010110 nell'AFN $\mathcal{N}$



AFN (Automi a stati Finiti Non deterministici)

Esempio: Computazione della stringa 010110 nell'AFN $\mathcal{N}$





Linguaggio accettato da un AFN

Definizione: Computazione e Linguaggio accettato da un AFN

Sia $\mathcal{N}$ un AFN e sia $w = w_1 \dots w_m \in \Sigma^*$.

- Siano $c_1 = (u_1 q_1 s_1)$ e $c_2 = (u_2 q_2 s_2)$ due configurazioni di $\mathcal{N}$. Allora

  $$c_1 \vdash_{\mathcal{N}} c_2 \text{ sse } q_2 \in \delta(q_1, u_2) \text{ e } s_1 = u_2 s_2.$$

  Indichiamo con $\vdash_{\mathcal{N}}^*$ la chiusura riflessiva e transitiva della relazione $\vdash_{\mathcal{N}}$.
- Un **passo di computazione** di w è una transizione fra due configurazioni c_i e c_j tale che $c_i \vdash_{\mathcal{N}} c_j$, $1 \le i, j \le m$.
- Una **computazione** di w è una sequenza di passi di computazione $c_1 \dots c_m$ di w tale che
  - esiste un $q \in F$ con $c_1 = (\epsilon q_0 w) \vdash_{\mathcal{N}}^* (w_m q \epsilon) = c_m$ e
  - per ogni j, $1 < j \le m$, si ha che $c_{j-1} \vdash_{\mathcal{N}} c_j$.
- w è **accettata** da $\mathcal{N}$ se esiste una computazione di w in $\mathcal{N}$.
- Il **linguaggio accettato** da $\mathcal{N}$ è l'insieme di stringhe accettate da $\mathcal{N}$:

  $$\mathcal{L}(\mathcal{N}) = \{w \in \Sigma^* \mid \exists q \in F \text{ tale che } \epsilon q_0 w \vdash_{\mathcal{N}}^* w_m q \epsilon\}$$

Corrispondenza tra AFD e AFN

Definizione: Equivalenza tra automi

Due automi si dicono *equivalenti* se riconoscono lo stesso linguaggio

Corrispondenza tra AFD e AFN

Definizione: Equivalenza tra automi

Due automi si dicono *equivalenti* se riconoscono lo stesso linguaggio

Teorema: Equivalenza tra AFD e AFN

Per ogni automa non deterministico esiste un automa deterministico ad esso equivalente

Corrispondenza tra AFD e AFN

Definizione: Equivalenza tra automi

Due automi si dicono *equivalenti* se riconoscono lo stesso linguaggio

Teorema: Equivalenza tra AFD e AFN

Per ogni automa non deterministico esiste un automa deterministico ad esso equivalente

Costruzione della trasformazione (AFN senza ϵ-transizioni)

Dato un AFN $\mathcal{N} = \langle Q, \Sigma, \delta, q_0, F \rangle$, costruiamo un AFD $\mathcal{A} = \langle Q', \Sigma, \delta', q_0', F' \rangle$ ad esso equivalente:

1. $Q' = \mathcal{P}(Q)$.
2. $\delta' : \mathcal{P}(Q) \times \Sigma \to \mathcal{P}(Q)$ tale che $\delta'(S, a) = \bigcup_{s \in S} \delta(s, a)$.
3. $q_0' = \{q_0\}$.
4. $F' = \{S \in Q' \mid S \cap F \neq \emptyset\}$.

Corrispondenza tra AFD e AFN

Esempio

Costruiamo un automa che riconosce il linguaggio delle stringhe su $\{a, b\}$ in cui il penultimo carattere è b:

- Costruire un AFN è immediato.
- Da questo possiamo costruire un corrispondente AFD.

Corrispondenza tra AFD e AFN

Esempio

Costruiamo un automa che riconosce il linguaggio delle stringhe su $\{a, b\}$ in cui il penultimo carattere è b:

- Costruire un AFN è immediato.
- Da questo possiamo costruire un corrispondente AFD.

		a	b
	$\emptyset$	$\emptyset$	$\emptyset$
0	$\{q_0\}$	$\{q_0\}$	$\{q_0, q_1\}$
	$\{q_1\}$	$\{q_1\}$	$\{q_2\}$
	$\{q_2\}$	$\emptyset$	$\emptyset$
1	$\{q_0, q_1\}$	$\{q_0, q_2\}$	$\{q_0, q_1, q_2\}$
3	$\{q_0, q_2\}$	$\{q_0\}$	$\{q_0, q_1\}$
	$\{q_1, q_2\}$	$\{q_2\}$	$\{q_2\}$
2	$\{q_0, q_1, q_2\}$	$\{q_0, q_2\}$	$\{q_0, q_1, q_2\}$